August 1982 Volume 1 $2.00

softalk

for the IBM Personal Computer

Robot On-Line

Exec Tecmar

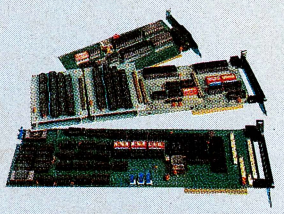
Peer Tutoring with the pc

First, the IBM Personal Computer,
The Next Step . . .

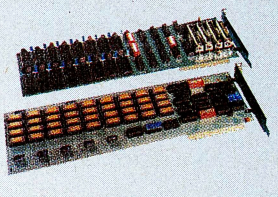
Tecmar

The TECMAR Expansion series is the first and only, complete line of expansion options available for the IBM Personal Computer.

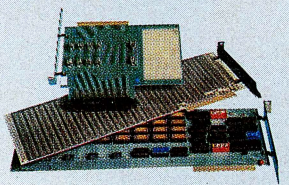
Now totaling over twenty-five separate options, the TECMAR series gives you the broadest range of expansion available for your IBM Personal Computer.



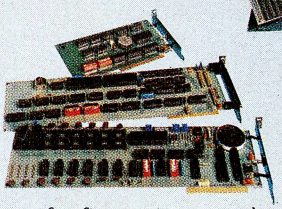
SYSTEM EXPANSION with a complete Expansion Chassis providing six additional system slots, a separate power supply and styling that complements the IBM system.



MEMORY EXPANSION in 64K 128K, 192K and 256K Byte increments of Dynamic RAM with parity. 32K Bytes of Static RAM, 32K Bytes of CMOS RAM with battery backup, or up to 128K Bytes of Read Only Memory.



PRACTICAL EXPANSION with two Serial ports and one Parallel port on a single board, or a Time of Day calendar with battery backup, a Voice Synthesizer with vocabulary in ROM and phoneme speech generation, even a BSR X10™ device controller for lights and appliances.



DISK EXPANSION through the addition of a five or ten megabyte Winchester disk. The disk options come enclosed in the TECMAR Expansion Chassis, providing additional expansion slots as well as Winchester disk storage. This approach assures you of unmatched system expandability for nearly any application.

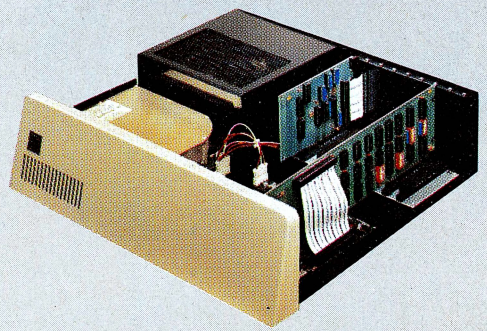
FUNCTIONAL EXPANSION is also available with TECMAR Speed Disk™ and print Spooling Software that give new functionality to memory options.

UNMATCHED EXPANSION for the serious IBM Personal Computer user through these and the many other TECMAR Expansion products available through participating COMPUTERLAND stores, and other fine computer retailers nationwide.

LABORATORY/INDUSTRIAL EXPANSION through an IEEE 488 interface, the Lab Tender with an 8 bit A/D and D/A, or the Lab Master for 12 bit A/D and D/A, a two axis Stepper Motor Controller, or the Parallel Digital-In/Digital-Out Base Board™, Video Digitization with Video VanGogh™.

DEVELOPMENT EXPANSION using an E+EEPROM programmer, Protozoa prototyping boards or a TECMAR Extender card.

NEW PRODUCTS are currently under development with many soon to be announced. At present shipping 26 unique IBM add-on products, we are still looking for needs to meet. If you have an need for a new product for the IBM Personal Computer, and would like to ask us about it, give a call on our Product Input Hotline at (216) 464-8317.



For IBM Personal Computer Expansion, TAKE THE NEXT STEP . . .

Tecmar Inc.

PERSONAL COMPUTER PRODUCTS DIVISION
23600 Mercantile Road, Cleveland, Ohio 44122
Telephone: (216) 464-7410 Telex: 241735

softalk

for the IBM Personal Computer

Features

Boards and Buses

A look at the Personal Computer's memory: How much do you need? How much can you really use?
Richard Kaapke 4

Exec Tecmar: The Expansion Team

How a small company became the biggest supplier of expansion hardware for the Personal Computer.
David Fitzgerald and Jim Salmons 14

System Notebook

DOS utilities—the methods and potential pitfalls of file manipulation.
Alan Boyd 18

A Robot Arms the Personal Computer

Robots aren't just science fiction anymore. In fact, IBM sells a model that can be programmed by your Personal Computer.
David Hunter 22

A Brainier Robot

A look at IBM's forthcoming robot, the sophisticated RS 1 Manufacturing System.
David Hunter 25

Stanford and Interactive Sciences: The Learning Chain

A summer workshop at Stanford introduces computers to children . . . and adults.
Jean Varven 30

Coaching Teachers

An IBM executive talks to educators about the present and future of educational computing.
Jean Varven 32

The Profit Plot

Using spreadsheets to deal with the complexities of inventory planning.
Shirley Fisher 34

Beginners' Corner

What is Basic? How programming languages mediate between computers and people.
Craig Stinson 39

Departments

Crosstalk 2
Puzzle Solution 2
Newspeak 8
Marketalk News 10
Marketalk Reviews 26
Tradetalk 41
Bestsellers 43

In Next Month's Softalk

Exec SofTech: an inside look at the company that distributes the p-System . . . an interview with Kenneth Bowles . . . IBM Pascal vs. p-System (Part 2) . . . "Getting Organized": how to organize data and utilities on your disks, how to use batch processing . . . and more.

Index of Advertisers

Amdek 3
Beaman Porter 27
Com-Pute Services 36
Computerized Management Systems 29
Continental Software ..... 5, 43, Cover 4
Corona Data Systems ..... Cover 3
DataMost 42
Information Unlimited Software 7
Insoft 37
Interface, Inc. 12
International Computers 35
Laboratory Microsystems 19
Microcorp 11
Micro Match 6
NF Systems 40
PC Connection 44
Personal Data Systems 9
Peter Norton 21
Quentin Research 33
Softalk for the IBM 41
Synergistic Software 13
Tecmar Inc. ..... Cover 2
TG Products 17
Yucaipa Software 25

Softalk Publishing Inc.
Chairman John Haller
Publisher Al Tommervik
Editor-in-Chief Margot Comstock Tommervik
Associate Publisher Mary Sue Rennells
Accounting Evelyn Burke
Editorial Manager Patricia Ryall

Softalk for the IBM Personal Computer
Editor Craig Stinson
Art Director Kurt A. Wahlner
Managing Editor Forrest Johnson
Newspeak Editor David Hunter
Guest Reviewer Peter Norton
Copy Editor Michael Ferris
Proofreader Harry McNeil
Contributing Editors
Financial Modeling Shirley Fisher
Hardware Richard Kaapke
System Software Alan Boyd
Production
Coordinator Elaine Cohen
Circulation
Subscriptions Pam Kelley
Deirdre A. Booth
Dealer Sales Donna Siebert
Advertising
West Coast Sales Jo Hilliard
Softalk
11021 Magnolia Boulevard
North Hollywood,
CA 91601
(213) 980-5074

East Coast Sales Paul McGinnis
Paul McGinnis Company
60 East 42nd Street
New York, NY 10017
(212) 490-1021

Midwest and Rocky Mountain Sales Ted Rickard
John Sienkiewicz
Market/Media Associates
435 Locust Road
Wilmette, IL 60091
(312) 251-2541

Cover photo by David Hunter.

Composition by Photographics, Hollywood, California. Printing by Volkmuth Printers, Saint Cloud, Minnesota.

IBM and Personal Computer are trademarks of International Business Machines, Armonk, New York.

Softalk for the IBM Personal Computer. Volume 1, Number 3. Copyright © 1982 by Softalk Publishing Inc. All rights reserved. ISSN: 0733-2173. *Softalk for the IBM Personal Computer* is published monthly by Softalk Publishing Inc., 11021 Magnolia Boulevard, North Hollywood, CA 91601; telephone (213) 980-5074. Second-class postage pending at Saint Cloud, Minnesota, and additional mailing offices.

Postmaster: Send address changes to Softalk/IBM, 11021 Magnolia Boulevard, North Hollywood, CA 91601.

Subscriptions: Complimentary trial subscription to all owners of IBM Personal Computers in the USA. If you own a Personal Computer, but you aren't receiving *Softalk for the IBM Personal Computer*, send your serial number and mailing address to Softalk/IBM Circulation, 11021 Magnolia Boulevard, North Hollywood, CA 91601. *Softalk for the IBM Personal Computer* is totally independent of International Business Machines. Back issues: $3.

Problems? If you haven't received your *Softalk for the IBM Personal Computer* by the fifteenth of the month, or if you have other problems with your subscription, Pam Kelley can help out. Call (213) 980-5074.

Moving? Send new address and old to Softalk/IBM Circulation, 11021 Magnolia Boulevard, North Hollywood, CA 91601; telephone (213) 980-5074.

crosstalk

Bestseller Controversy

I have just purchased your premiere issue from Computerland in West Hartford, Connecticut. As a brand-new IBM pc owner, I was especially happy to see Beginners' Corner. I have already used some of the information offered through this helpful column. I also enjoyed Bestsellers. This helps in making decisions of what software to buy and what must be working. All of these things are very helpful to a new computer owner.

Mari Bentley, East Hampton, CT

I think your Top Thirty is a very weak feature. I cannot think of why I'd want to know which program is the most popular. Besides, your polling technique biases the survey in favor of software sold through retail outlets. Think of how misled a person might be by the information that *EasyWriter* is the best-selling program. It is well known that *EW* is one of the worst pieces of software available.

I very much appreciated your feature on user groups (July 1982). I am working hard to organize an IBM pc user group here in northern Orange County, California. I would very much appreciate the inclusion of my name and address in any future directory you may decide to publish.

Glenn A. Emigh, 1533 Sherwood Village Circle, Placentia, CA 92670

Reviews for Marketalk

I am very interested in reading your reports and evaluations of software. Many of us had to learn the hard way in purchasing useless software that could not be returned.

Frank P. Vlamings, Newark, CA

Thank you for your invaluable service. I've been reading your magazine long before the IBM pc came along, since I also own an Apple II Plus. I feel that your publication is one of the finest, especially since choosing the right software or hardware happens to be quite difficult because of the many quality products in this type of market.

Luis G. Gomez, New York, NY

If I may interject a bit of a request for future editorial content, I personally am interested in seeing software and hardware reviews, along with helpful how-to articles, with the emphasis on utility routines. I hope that this type of article will make up a major portion of the magazine.

Michael J. Tracy, Detroit, MI

I truly hope that *Softalk for the IBM* will not just be an endless series of product reviews and gadgets. Personal computers in general have such an incredibly rich potential for education and involvement that one could spend years with nothing more than the basic setup and a programming language.

William C. Lukes, Aspen, CO

Service Headache

According to the IBM Product Center in Orlando (where I bought my pc), signing up for IBM service after the ninety-day warranty has expired requires an on-site visit by IBM. The charge for such a visit? According to the Product Center manager here, about $100. I found this by contacting IBM for service 111 days after delivery. IBM will not contact you during the warranty period regarding maintenance contracts like they do with typewriter purchases—you must take the initiative. Owners of new machines—keep track of those ninety days!

Michael Truffer, Deltona, FL

Thanks for the Support!

I purchased your first issue a few days ago and enjoyed all of the material. It was a major factor in convincing me to go ahead and get a pc at this time.

Chuck Reese, Lawrence, KS

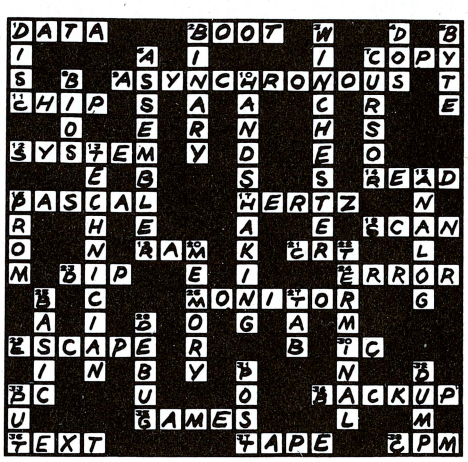
I found the premiere issue supportive and it has already added measurably to my use of the pc.

Bill Jensen, Sparta, WI

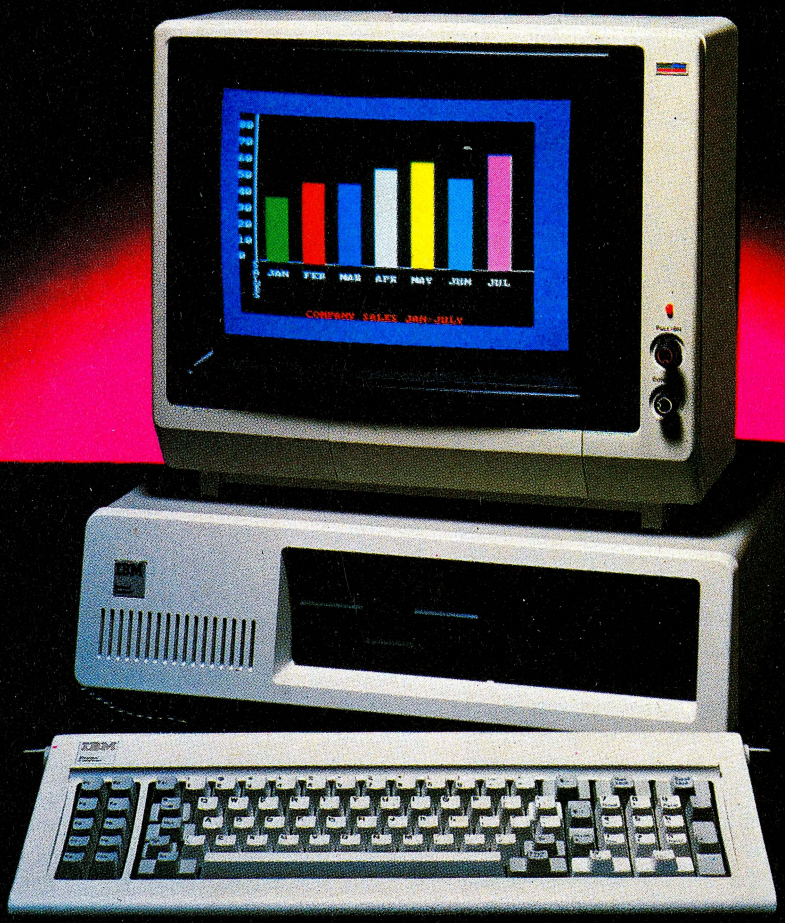
Picked up your premiere issue along with my IBM pc. Looks like we both have winners.

Richard S. Manchee, San Ramon, CA

Solution to Last Month's Puzzle



Your IBM personal computer deserves an Amdek Color-II monitor



To take maximum advantage of all your new IBM Personal Computer can do, you need a compatible monitor with extensive capabilities of its own. And for that, there's simply no better choice than Color-II from Amdek.

Color-II's high-resolution colors can help you visualize and analyze the most complex data. Whether you need your IBM for word processing, graphics, business analogy or scientific applications, Color-II has the display technology you need to utilize your IBM to the fullest.

Color-II features include:

- RGB input for high-resolution graphics and more vibrant colors.
- 80 x 24 character display.
- 560(H) x 260(V) resolution.
- Intensification modulation for 16 color capability to match IBM's high resolution colors.
- IBM plug compatible cable and connector.
- Molded-in carrying handle for easy portability.
- Front-mounted controls for quick adjustments.
- Complete with one-year warranty covering all parts and labor; two-year warranty on picture tube.

So for the most efficient, effective use of your IBM computer, ask your dealer about the remarkable Color-II—part of Amdek's complete line of color, green phosphor and black & white video monitors. Then match Color-II's performance and price against any other high-resolution monitor. For quality and value, you'll choose Amdek.

AMDEK

Amdek Corporation, 2420 E. Oakton St., Suite E, Arlington Heights, IL 60005. (312) 364-1180 • TLX: 25-4786

BOARDS & BUSES

by Richard Kaapke

The IBM Personal Computer is probably the first computer intended for personal use that has the capacity to address and use over a million bytes of memory. How does the computer use all that memory space? What good does it do you? Do you really need that much memory?

A computer's memory defines the boundaries of its capability. Typically, a computer with a large memory can run programs that one with a smaller memory cannot.

Mapping Memory. The System Memory Map from page 2-25 of the IBM *Technical Reference Manual* and reproduced as figure 1 shows how memory is organized in the pc. IBM has divided the available memory space into four major partitions.

The first partition, covering the addresses 0 to 262143, includes the memory on your system board (the main printed circuit board of the computer) and up to 192K of add-in memory. You can have as little as 16K and as much as 64K on the system board. The term add-in memory refers to memory on cards that you plug into any of the Personal Computer's five peripheral slots. The System Memory Map calls this memory "I/O Channel Added Mem."

A full system board plus 192K of add-in memory would give you a total of 256K, which the manual claims is the "present system max."

The second partition is marked "384K Memory Future Expansion." We'll have more to say about this in a moment.

In the third partition, from address 655360 to address 786431, are two small spaces taken up by the display adapters—monochrome and color/graphics. Each of these adapters has its own memory space in the 128K total reserved for graphic and/or display buffers. These spaces are the specific locations where you'll find the built-in semiconductor memory that comes with these adapter cards. These memory buffers store the text or graphics produced by the Personal Computer.

The fourth partition, addresses 786432 to 1048567, is marked 256K ROM Address Space. In its present form, the IBM Personal Computer uses only the top 40K of this memory region. That 40K includes the Basic Interpreter and the Power On Self Test (POST) programs.

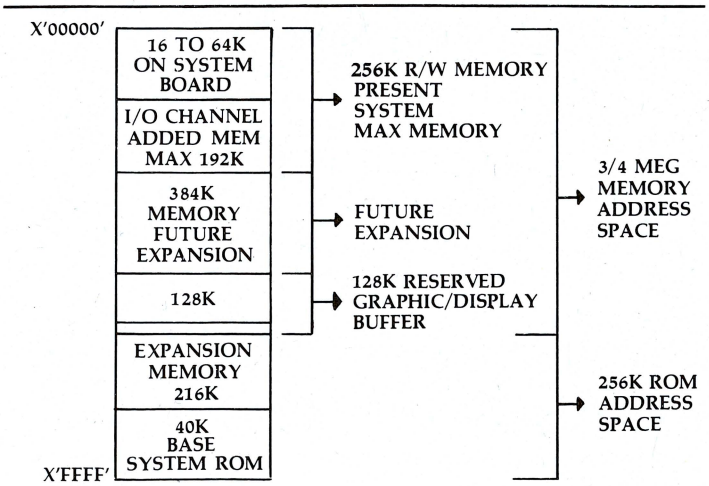


Figure 1. System Memory Map. (Courtesy of IBM Corporation)

Back to partition two. Currently, IBM advertises its Personal Computer as a machine with a 256K capacity, although the System Memory Map shows a "future expansion" area extending to 640K. Clearly, the Personal Computer is designed to be—either now or in the future—a machine that can use 640K of semiconductor memory.

So why doesn't IBM claim it to be more than a 256K machine? For one thing, IBM's own add-in memory cards come in only one size—64K. Assuming that two of your five peripheral slots are taken up by a disk drive and a display adapter, you'd have three slots left over for memory expansion. That means that the maximum amount of IBM-made memory you could add to your system, at this time, would be 192K. That plus 64K on your system board would give you a total of 256K.

At present, IBM only tests its "big memory" software on systems that max out at 256K. If they were to test software on larger systems, they'd have to use non-IBM boards.

Can you in fact go beyond the alleged 256K maximum? Emphatically yes. A number of companies, other than IBM, make IBM plug-compatible memory cards that go to 256K or more per card; some companies have cards with as much as 512K. The System Memory Map says—clearly, if not loudly—that you are permitted to extend your memory as far as 640K.

The Dip Switches. There's another consideration, however. Inside the system unit are a couple of dip switches, with which, among other things, you tell the computer how much memory you have in your system. Page 2-30 of the *Technical Reference Manual* and page 5-16 of the *Guide to Operations* both illustrate the use of these switches.

Switch 1 is used to indicate the amount of memory you have on your system board. If you have 64K on the system board, positions 3 and 4 of switch 1 should be set to off.

Positions 1 through 4 of switch 2 indicate how much additional memory you have. The status of these four positions represents a four-figure binary number that tells the computer how many multiples of 32K you have—in addition to the 64K on your system board. The manual only illustrates the appropriate switch settings for up to 192K of add-in memory. But a four-position binary number can represent any value from zero to fifteen, so it's easy to see that switch 2 could be set to indicate as many as fifteen multiples of 32K.

Fifteen increments of 32K added to 64K on the system board yields a total of 544K. So it's safe to assume that your Personal Computer could use as much as 544K.

You may be wondering if your programs—your own or those you buy—will be able to address that much memory. Typically the answer is yes. A program either will be restricted to 64K or will be able to address a full megabyte of memory.

Why 64K or 1M and nothing between the two? Simply put, a program may either address memory through a single "offset"

1st Class Mail™ It does everything but lick the stamp.

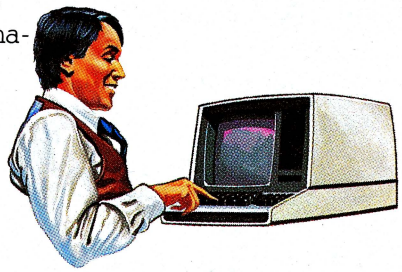
We've made a mailing list program worthy of today's remarkable microcomputers. It's called *1st Class Mail*™—but you can use it for anything you'd keep on a card file, like your car's servicing history or your address book. And it's not only the most flexible mail list/utility data base you can buy—it's also incredibly easy to use.

You create the label.

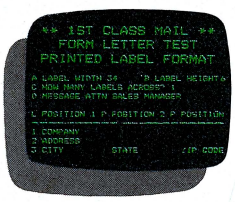
Since you're not stuck with the same old name/company/street/etc. format, you can make the label look like whatever you want. And with the new feature that lets you print right on your envelopes, you not only save time—you give your correspondence a very professional look.

Super search and sort.

1st Class Mail™ lets you search and sort in any of 12 different categories. So you can organize your mailings by zip code (up to 10 digits), alphabetical order, state, company, any way you want. Fast. And if you're using *1st Class Mail*™ for another type of data base, like a record collection, you can sort by whatever categories you choose— from String Quartets to Rhythym and Blues. It's extremely flexible.



Personalized entries.

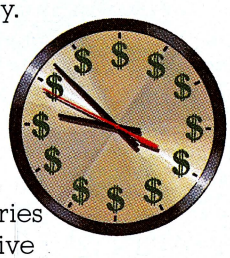


One of the ways we've made *1st Class Mail*™ especially "user friendly" is by letting you create your own headings for each entry line. So your computer speaks *your* language.

The form letter feature.

1st Class Mail™ is compatible with many word processors, such as Screenwriter II,™ Applewriter II,™ Applewriter III,™ Pie Writer,™ and WordStar/Mailmerge.™ Just a few simple commands and you can combine your form letters with your mailing list for professionally formatted documents. All automatically.

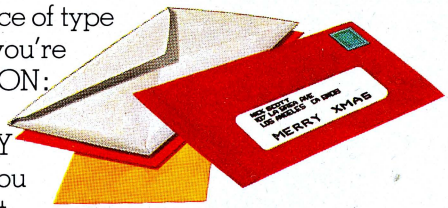
Automatic entry.



If time is money, this feature translates minutes directly into dollars. It automatically includes, or searches for, particular categories in each entry you record. Repetitive typing and sorting becomes a push-button joy.

The special message line.

Now you can put a special message anywhere on the label—in a choice of type styles. Whether you're saying ATTENTION: SALES MANAGER or MERRY CHRISTMAS! you can do it automatically. And right on the envelope.



Stop by your Continental Software dealer. Or call us today to find out all about *1st Class Mail*.™ No matter what you're cataloging, from a household inventory for insurance purposes to a 25,000 person mailing list, *1st Class Mail*™ is the best program you can buy.

It's true, you still have to lick the stamp.

But we're working on it.

Available for: Apple II,™ Apple III™/III with Profile,™ IBM-PC™/IBM-PC with Tecmar™ hard disk/IBM-PC with Davong™ hard disk.

Apple II/II +, Applewriter II, Applewriter III are registered trademarks of Apple Computer, Inc. Screenwriter II is a registered trademark of On-Line Systems. Pie Writer is a registered trademark of Hayden Book Co. WordStar Mailmerge is a registered trademark of MicroPro, Inc. Tecmar is a registered trademark of Tecmar, Inc. Davong is a registered trademark of Davong Systems, Inc. IBM-PC is a registered trademark of IBM Corp.

Continental Software

Continental Software, 11223 South Hindry Avenue, Los Angeles, California 90045, Telephone (213) 417-8031

register that is sixteen bits wide (allowing it to specify a maximum of 65,536 addresses) or it may address memory by means of two registers—a segment register arithmetically added to an offset register. In the latter case the program would have a total of twenty bits available for addressing, which means it would be able to specify 1,048,576 unique addresses.

Only by specifically "clipping" or coding software can a programmer restrict memory addressing to some intermediate maximum, like 256K. Has this been done in the IBM software packages? Initial experiences say no—and for now IBM will say only that they've tested their software with a maximum of 256K.

If you want to use an IBM software package on a system with more than 256K, you have no guarantee from IBM that the package will take advantage of the memory in excess of 256K. Note, however, that software from other sources, who do promise support of more than 256K, is starting to appear in the market.

What Good Is It? By now you may be asking what good all that extra memory is. The answer's not simple, but here are some points to consider:

1. Many new programming languages now require more than 64K to run. Often this is true of the compilers offered by IBM in its Computer Language Series for the Personal Computer—languages such as Fortran, Cobol, and Pascal. Some of these languages will work in 64K but perform much better in systems having larger memory.

2. Programs will appear in the marketplace that will not run in less than 256K. Most notable of these at the moment is Context Management Systems's comprehensive business package, *MBA*.

3. Certain programs store in memory all the data they're working with. It's difficult to work with these programs on small-memory systems. *VisiCalc* is a good example of such a program.

4. Concurrency is on the way. When Digital Research's Concurrent CP/M-86 becomes available, it will be a big advantage to have lots of memory, because you'll be able to have several programs in memory at once, all (apparently) running simultaneously.

Parity Checking. One other point should be made about IBM and IBM-compatible memory. Semiconductor RAM is designed so that once a pattern of zeros and ones is written to a memory address, that pattern remains fixed until another write operation is performed at the same address or until the power is turned off. To catch any unwanted changes in values stored in memory, your computer uses a system called *parity checking*.

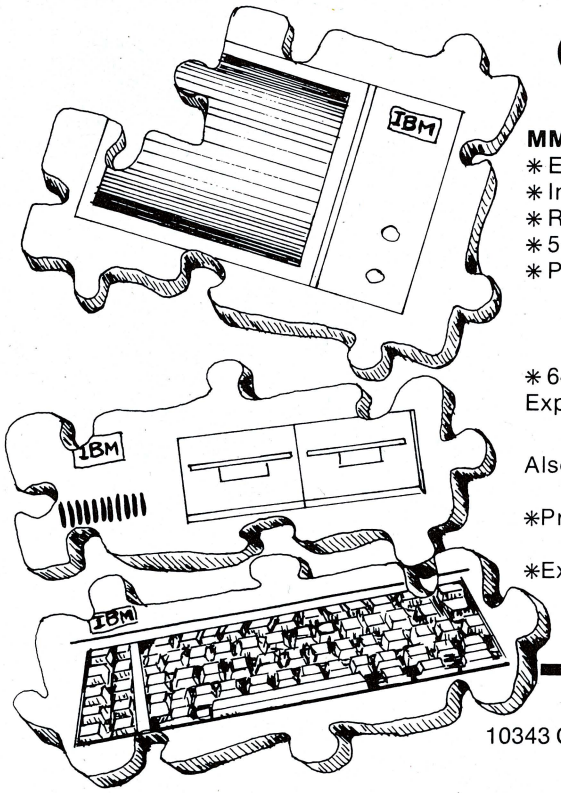
For every eight bits of data stored, the computer records another value in a ninth bit. This ninth bit, called a parity bit, is used solely for error checking. Its value—one or zero—depends on the value stored in the other eight bits. If there are an odd number of ones stored in the other eight bits, the parity bit will be set to zero. If the eight data bits hold an even number of ones, the parity bit will be set to one. In either case the total number of ones in the nine bits will be odd. This technique is sometimes called *odd parity*.

When the computer performs a read operation at a given address, it counts up the number of ones in the eight data bits and corresponding parity bit. If that number isn't odd it knows that an error has occurred. The error may have resulted from an unwanted change in the parity bit, an unwanted change in one or more of the data bits, or from unwanted changes in both data and parity bits. In any case, you've got a problem.

The computer will respond by generating an I/O Channel Check, and that ultimately will result in the appearance of an error message on your screen. Among other things, your screen will display a four-digit code indicating which circuit has failed. At that point you won't be able to operate your computer any further until you turn it off and turn it back on again.

Parity errors shouldn't be an everyday occurrence in your Personal Computer. But chips do fail, and it's nice to have this early warning system in case they do begin to go. ▲

Micro Match solves the IBM PC expansion puzzle.



MM64 MEMORY BOARD
* Expands in 64K increments
* Includes sockets for all RAM
* RAM test program included
* 5 year warranty
* Parity .... MM64-1 (64K) $400.
MM64-2 (128K) $510.
MM64-3 (192K) $620.
MM64-4 (256K) $730.
* 64K Expansion Kit (9 chips) Expands MM64 memory ....$108.

Also available:

*Prototyping Board .....$50.

*Extender Board .....$45.

TANDON DISK DRIVES FOR YOUR IBM PC.
* Installation instructions included
* IBM PC compatible

TM100-1 (Single Sided 160K byte storage) .....$225.
TM100-2 (Double Sided 320K byte storage) .....$295.

* Send for our free Catalog

To order: Send check or Money Order (US only). Add $3.50 for Shipping and Handling. California residents add 6% sales tax. Master Card/VISA accepted.

MICRO MATCH *Quality Products*
Personal Computer

10343 Commerce Ave. Dept. S • Tujunga, CA 91042 • (213) 353-5929

It's easy.

Easy to use.
Easy to afford.
Now.
The state-of-the-art in spell-checking.
EasySpeller™ from IUS
88,000+ words / 92K disc space
Exclusively, for the IBM personal computer.

EasySpeller™

Extraordinary.

Information Unlimited Software, Inc.
2401 Marinship Way
Sausalito, California 94965
(415) 331-6700

Extraordinary software.

newspeak

Δ **Maybe Next Year.** According to Coopers & Lybrand, the world's largest accounting firm, there are three kinds of favorable tax treatment available to buyers of computer software as the result of the Economic Recovery Act.

The Los Angeles–based firm says the most aggressive method is to treat the costs of purchased but customized software as a current expense. If it can be shown that the risk of the usefulness of the software is borne by the taxpayer, such costs are deductible for tax purposes.

You may also be entitled to a research and development tax credit for the cost of software if you capitalize the amount and amortize it over the useful life. The IRS has attempted to treat computer software as an intangible asset if bought separately from hardware. Intangible assets are eligible for amortization over a short life, as opposed to being depreciated over five years, under the tax laws.

If the software can be classified as tangible personal property, an investment tax credit and a five-year depreciable life are available. To accomplish this, off-the-shelf characteristics must be identified—the software must have been purchased as a standard program package.

Personal computer owners as well as businesses should be able to take advantage of these tax benefits. David Oifer, a partner in the firm, maps out a course of action: "It clearly is in the interest of the taxpayer contemplating purchase of computer software to review the tax situation involved and incorporate tax planning into the contractual agreement with the software vendors."

Δ **Bencher Liberation.** Before *elegant* became another buzz word for *expensive* when applied to cars, watches, and condominium living, the word was (and still is) a mathematical term for precision and neatness.

Elegance is more than just a word in the world of computer design engineering; it is a necessity. If anything needs to be elegant today, it is the tiny chips and circuits that populate the personal computer.

Achieving a refined gracefulness of concept comes only after much hard work. The pursuit of elegance is a combination of experience, intuition, and persistence. It can often result in a moment of Zenlike satori, or enlightenment, as engineering solutions seem to appear magically on a complex schematic drawing.

All this—providing the designers aren't too involved with tedious clerical work to notice. Theirs is a world of reference manuals, standard formulae, and specification sheets. A lot of pencils get pushed as designs are manually checked for errors, corrected, and modified, usually a three to four week process. At the bench level of engineering, between the logic capture, the documentation, and the calculation, who has the time for elegant thinking?

Today's computer market demands competitive innovation: companies are pressed to be the firstest with the mostest. This is making products more complex than ever before, yet they seem to have a shorter life span. Second-generation products are released seemingly before the first are settled on the shelves.

Computers themselves have been gradually taking over the back end of the computer design task during the last decade. Developing prototypes and manufacturing are two areas man has gradually given over to the machines. The front end of the design business, at the bench where the engineer works, is the next area scheduled to be opened up to microtechnology, freeing the designers to do what they do best: design. Spending less time calculating and capturing logic, they'll spend more time thinking through implications, analyzing their work, considering alternatives. . . .

In other words, thinking elegantly.

The engineering system that can do this job has been introduced by Mentor Graphics of Portland, Oregon. Called the Idea 1000, it's the first totally integrated computer-aided engineering system. Its credentials: local processing and mass storage at each work station, interactive simulation, and high-speed local networking with a distributed database. Logic designs can be created, captured, analyzed, verified, and documented by computer—tasks previously done by hand.

All of this can now be done before any prototype is constructed, and done in minutes and days instead of days and weeks.

The Idea 1000 system consists of an Apollo Domain computer, its Aegis operating system, and six application software packages. It can be used as a stand-alone engineering station or as part of a network of as many as 200 separate stations. The system's power grows with each new addition, unlike most micro or mini processor networks.

The applications programs are: *Structured Logic Design*, an intelligent graphics editor that creates and captures logic diagrams; *Interactive Logic Simulation*, which can build designs either top-down or bottom-up; *Project Communications*, with electronic mail capabilities; *Document Preparation*, which integrates text and graphic information; *Outputs*, which transfers designs to other systems or simulators; and a custom application program that allows designers to write their own analysis routines.

Statistics show that design engineers only spend 50 percent of their day actually designing. The other half is taken up with paperwork. The Idea 1000 looks as if it's going to alter this ratio radically.

An elegant solution, don't you think?

Δ **Pass the Chips.** According to some estimates, sales of 64K dynamic RAM chips will reach one billion units per year by mid-decade. Not quite an industry standard yet, 64K RAM chips posted about $100 million in sales during 1981.

Intel has just announced the release of its second-generation 64K dynamic RAM, the 2164A. The company's first-generation product was withdrawn from the market because of difficulties encountered in certain memory system applications. Intel decided to go ahead with the second-generation product rather than spend money to correct the first one.

Intel's innovative use of redundancy is one of the most interesting features of the 2164A. Basically, a redundant device has extra columns or rows. When leading-edge technology products like the 2164A are made, any minor extraneous impurity in the basic silicon wafer can cause failures to occur in specific bit areas. Redundancy eliminates the requirement to dispose of an entire chip when there's a processing error in a single bit. Because extra rows and columns exist, a bad bit location can be disabled and a new site included.

The 64K chips are expected to be the largest memory market in history. In 1977, 4K RAM chips hit their peak sales of $145 million. The peak year is projected to be 1982 for 16K chips, with sales close to $500 million. The acceleration progression gains emphasis when you know that Intel introduced the first 1K dynamic RAM in 1970. Now, industry-wide sales of 64K dynamic RAM chips are estimated to peak at $1.5 billion in 1985 or 1986.

Δ In the old days, not too many years ago, software was something that came on a cassette, and you got it by trading your programs for your friend's or by mail from a computer company. An individual program might be good, bad, or mediocre, but the

documentation usually fell into one of the latter two categories because it was written by the programmer (who frequently ran the company as well). Programmers were likely to be much more literate in Basic than in English and tended to assume you had the same expertise that they had.

These days, software is something that comes on a disk. You get it from your local computer store, which stocks a variety of programs and has trained personnel to assist you. The quality of the documentation has improved somewhat, but we are still a long way from being able to buy a program with the ease and confidence of buying a stereo record. Prices are high, quality is uneven, and the average buyer is reluctant to risk his cash without the advice of a computer professional.

While most software companies can only look hungrily over the fence, a few have been able to break into the mass market. The video game companies are in the forefront.

Owners of Atari video game computers can find programs on the shelves at more than twenty thousand outlets, including drugstores, department stores, and toy stores. Atari has made the game computer a commonplace object, like the toaster or transistor radio, and other companies are following up on its success.

Of course, it is one thing to plunk down $10 for a copy of *Home Run* and another to lay out $250 for *VisiCalc*. Nevertheless, it may not be long before you can buy software for user programmable computers at your local bookstore. Lifetree Software is test marketing its *Volkswriter* program through the Ingram Book Company, a major book distributor that has carried Atari cartridges since April. Lifetree president Camilo Wilson thinks that in the future computer programs will be treated like any other product, provided they don't require support from the dealer. "You can't carry high-service items," he says. "If you have a complicated database program, you're out of luck."

Several companies apparently believe that word processors will be the programs that lead the industry into the general market. Jim Edlin, former editor of *PC* magazine, has founded the company Bruce & James Program Publishers with the specific purpose of selling to bookstores. The first program will be *Wordvision*, a word processor scheduled for release in October. *Wordvision* is intended as a simple, entry-level program suitable for the mass market. Its biggest attraction is the price: less than fifty dollars. Jumping on the bandwagon, a company called Design Enterprises is offering for 60 percent of that *Word-Worker*, a word processor written in Basic.

Not surprisingly, as software houses look hungrily over the fence, they sometimes see book publishers and distributors looking hungrily back. For example, B. Dalton Booksellers is currently test marketing software in some of its stores. If the mammoth B. Dalton chain decides to distribute programs, other companies are sure to follow.

The book-program connection may seem weak to some, but to many book publishers it seems natural. For example, Prentice-Hall has been selling technical books for years through its subsidiary Robert J. Brady Company. When they recently produced a programming book, it was easy to enclose a disk and market the package through the same channels as before.

Prentice-Hall's general publishing editor, Gregory Roench, thinks the company has a big advantage in selling to bookstores. "The typical ma-and-pa computer business is stuffing programs in plastic bags and sending them out cash on delivery. We can offer the bookstore better packaging and better terms."

On the other hand, there are definite limits on selling to bookstores. "The typical bookstore is not prepared to invest any time and money in training personnel. Neither is it prepared to invest capital in putting a computer on the floor. It has to be stuff that does not require maintenance and does not require demonstration."

As the computer market expands, there will be a broader market for programs and broader merchandising to keep up with it. Roench predicts, "I think you'll be seeing programs in bookstores in two years." ▲

Rams & Much Much More!

for IBM Personal Computer

Available Now

PACK-RAM™ $225.00 & up

* Increase system memory up to 256KB.
* Sockets provided for user up-grades.

PACK-COMBO™ $175.00 & up

* Combination of RS232 serial port, a battery backed up real time clock and an optional parrallel printer port.

PACK-RAM + COMBO™ $435.00 & up

* Combination of PACK-RAM™ and PACK-COMBO™ on a single card for space savings.

Coming Soon

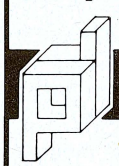
PACK-COMM™

* It will transform the PC into a Voice & Data Communication terminal even without operator's attention.
* Bell 103J compatible modem is optionally integrated on the card with auto answer & auto-dial capability.

PACK-HDK™

* Externally provided 5 Mega Byte formated on line storage of either fixed disk or removable cartrige disk.
* Compatible to either PC-DOS or CP/M-86 operating systems.

All products use one card slot, burn-in tested, and carries 1 year warranty.



PERSONAL DATA SYSTEMS

1110 Wrigley Way • Milpitas, California 95035
(408) 262-7880

Orders Accepted Saturdays 9 a.m. - 4 p.m. Also

marketalk news

Unless otherwise indicated, software listed runs in DOS on machines with either display adapter and requires 64K and at least one disk drive.

Δ Last month's Marketalk News reported three programs, *Print Graf, MXPlus,* and *Millionaire,* all from **Micro-Z Applications.** We erred. *MXPlus* and *Millionaire* are indeed the products of Micro-Z Applications (22704 Ventura Boulevard, Woodland Hills, CA 91364; 800-835-2246). *Print Graf,* on the other hand, is marketed by **Micro-Z Electronic Systems** (Box 2426, Rolling Hills, CA 90274; 213-377-1640). We regret any inconvenience this may have caused the companies involved.

Δ *T.I.M.*, from **Innovative Software** (9300 West 11th Street, Overland Park, KS 66210; 913-383-1089), is an abbreviation for Total Information Management. The program offers eight field types, including date fields and calculated fields; files may be protected with four-character passwords and inspected in a number of ways, including sequential, sorted, random, absolute, or relative orders. Up to forty key fields may be specified. Requires eighty-column display. $495.

Δ **USI International** (71 Park Lane, Brisbane, CA 94005; 415-468-4900) offers a monochrome display in amber, the color that's been legislated as standard in several European countries. Called the Pi-3, the twelve-inch monitor has a 20MHz bandwidth and is said to be suitable for eighty-column display. Requires color/graphics adapter. $289.

Δ **Peerless Engineering Service** (5819 Soquel Drive, Soquel, CA 95073; 408-462-0330) offers a number of commercial graphics programs. *Coordinate Geometry* combines an intersection solution program with a planimeter that boasts an accuracy to 1:1,000,000. Requires color monitor and asynchronous card. $95. Δ *PC Map* is a topographical plotting program that can input data directly from a surveyor's stadia traverse field notes. Requires color monitor and asynchronous card. Two drives recommended. $995. Δ *Lettering and Calligraphy* can draw letters in four fonts—simplex, roman, italic, and gothic. Requires color monitor and asynchronous card. $95.

Δ **Little Bit** (469 Edgewood Avenue, New Haven, CT 06511) has a *Floppy Disk Librarian* that automatically updates information by reading directory tracks. $39.95.

Δ **International Computers** (13048 Olive Boulevard, Saint Louis, MO 63141; 314-878-8838) offers two Forth programs: *SerPrinSoft,* which makes serial printers work as they are supposed to ($79.95), and *Mr. Lister,* a mailing list program compatible with *WordStar* and *MailMerge* ($79.95).

Δ From **Single Source Solution** (2699 Clayton Road, Concord, CA 94519; 415-680-0202): *PCWord 2.1,* a full-screen, menu-driven text editor. $49.95. Δ *PCForms 2.1,* a text-formatting system that supports macros nested ten-deep. $49.95. Δ *PCScreen 2.1,* a screen input format generator. $49.95.

Δ *TextPlus* is a word processor with full-screen editing from **Owl Software** (1605 East Charleston Boulevard, Las Vegas, NV 89104; 213-982-6243). It is designed to handle such things as business letters, mailing labels, and phone lists as well as regular correspondence. $200.

Δ **Microcorp** (913 Walnut Street, Philadelphia, PA 19107; 215-627-7997) has released *IBM Comm-Plus,* an expanded communications package that supports uploading and downloading, autodialing (with Hayes modem), automatic sign on, and other enhancements to IBM's asynchronous communications package. $25.

Δ **MicroPro** (1299 Fourth Street, San Rafael, CA 94901; 415-457-8990) is releasing a new version of *WordStar* that is compatible with PC-DOS 1.1. Owners of earlier *WordStar* versions can have their disks updated by returning them to the dealer.

Δ PC-DOS Version 1.1 is not compatible with hard disks from **Davong Systems** (1061 Terra Bella Avenue, Mountain View, CA 94043; 415-965-7130). However, Davong is mailing a software fix to every owner who sent in a registration card. Other owners can get the necessary information from their local dealers.

Δ **IBM** has reduced the price of its 64K memory board. The price at IBM Product Centers is now $350, down from $540. Prices at other pc dealers may differ.

Δ The Smartmodem 1200, from **Hayes Microcomputer Products** (5835 Peachtree Corners East, Norcross, GA 30092; 404-449-8791), is a 300- or 1,200-baud Bell 212A compatible modem suitable for both pulse and touch-tone dialing. $699.

Δ *BankReporter I* is a comprehensive set of *VisiCalc* templates for commercial bank applications. Models cover planning, reporting, and asset/liability management for commercial banks. Requires 256K, 132-column printer, and *VisiCalc.* From **Distributed Planning Systems** (16133 Ventura Boulevard, Encino, CA 91436; 213-992-4447). $5,000. Rental, $225 per month.

Δ **Ticom Systems** (13470 Washington Boulevard, Marina del Rey, CA 90291; 213-827-7118) has announced the availability of *Final Copy,* an integration of word processing, forms processing, remote communications, and data management functions. Double-sided drives recommended. $495.

Δ *Personal Mailer,* from **Computer Age of San Francisco** (825 Masonic Avenue, San Francisco, CA 94117; 415-921-7792), offers password protection; twelve user-defined fields that may be searched, sorted, or printed; and the choice of one-, two-, or three-abreast label printouts. Requires eighty-column display. $65.

Δ **Amdek** (2420 East Oakton Street, Arlington Heights, IL 60005; 312-364-1180) manufactures a four-color x-y plotter, the Model DXY. It has a 10-by-14-inch effective plotting range, a maximum plotting speed of 2.73 inches per second, and a step size of 0.003-inch. A Centronics interface is furnished for connection to your pc. $949.

Δ **Leading Edge Products** (225 Turnpike Street, Canton, MA 02021; 800-343-6833) has introduced a twelve-inch green-screen monitor suitable for eighty-column display. $99.

Δ *The Personal Computer Journal* (West 2317 Garland, Spokane, WA 99205) is a monthly magazine on disk. It includes tutorial articles, utility programs, and games. $10 per issue, $85 per year.

Δ **Strategic Simulations** (465 Fairchild Drive, Mountain View, CA 94043; 415-964-1353) offers *The Warp Factor,* a space warfare game for one or two players that emphasizes detailed ship handling and strategy. $39.95.

Δ *Infotory* is an inventory management program from **SSR** (1600 Lyell Avenue, Rochester, NY 14606; 716-254-3200). It is menu-driven, has a flexible reporting feature, and can withhold display of confidential information. Requires two disk drives. $425.

Δ **Softrend** (Box 1462, Charlottesville, VA 22902; 804-293-5344)

TOOLS FOR THE IBM PC

MICROCORP

913 Walnut Street, Philadelphia, PA 19107 215/627-7997

Source TCM033, Compuserve 70335, 1027

SAVE YOUR SLOTS!

Seattle 64K Memory/Communications Board: Saves 4 slots over IBM. Expandable.
Board . . . . . $475. Each 64K Addition . $200
PLUS—$100 Credit from MicroCorp for IBM Communications boards sent with order.

MicroSoft 64K Memory Card/Software Package: Just announced! RAM Drive option lets you copy entire disk into machine memory. Read/write at hard disk speed: 50 times faster than a floppy! Software supports other systems as well. Board/software . . $495
Each 64K addition $200

WORDSTAR WORDSTAR WORDSTAR

Now available: MicroPro's WORDSTAR! Full printer controls for most standard printers, handles documents larger than memory—All the features that have made WORDSTAR the world's best-selling word processing program. PLUS: Full support of IBM/PC (cursor controls, Function keys, horizontal scrolling and much more) $495
Optional Mail/Merge for maillists $150
PLUS—$85 credit from MicroCorp for EASYWRITER when sent with order!

COMMUNICATIONS

IBM COMM-PLUS
Now your IBM Communications software can do even more. Complete on-disk patch kit lets you modify your program to support: Upload/Download (send prepared text or capture online information), printers, auto-dial for Hayes Stack modem, automatic single key (FB) sign-on $25
(Includes Intelliterm Communications Tutorial for beginners. Full credit when you buy soon to be released Intelliterm.)

INTELLITERM
The premier communications program. Fast, flexible, incredibly easy to use. Unique split-screen mode. High-speed system-to-system-transfer. Much more!
(7782) $150
IBM Communications Cards (pre-owned) . . . $125

SOFTWARE TOOL KIT

SUPERZAP $40
View and modify any byte of a PC DOS diskette. Access by absolute track, sector number, or filename. Move track-by-track or sector-by-sector, up and down. Ideal for patching ruined directories. Note: NOT intended for copying protected software.

VIEWDISK $20
Complete disk exploration and display. Graphic view of storage. Complete documentation on how PC disk files are stored and how to alter directory. Lists and prints directory alphabetically by name or extension, creation date, sizes, file attribute. Ideal for hard disk users.

RECOVER $20
Yes there is life after accidental death. Recovers erased files.

ATTRIB $10
Modifies files to/from Invisible, System.

Available singly or as package.
PACKAGE PRICE: $80. Includes free CLOCK program (time, date, elapsed job time).

PASCAL Tools
Selected PASCAL utility programs, on disk, from Kernighan/Plauger's "Software Tools in PASCAL." Includes 366 page book $75

DATABASE

T.I.M. III—State-of-the-art, user-friendly data mgt. program. Up to 8 field types; 32,000 records per file; 40 fields per record; 40 sorted orders; and more. Order now and receive complete Software Tool Kit, a $90 value, free! $500
DATAMASTER—see Jan, Mar PC ads $800

UNBASIC

Powerful BASIC preprocessor for maintaining modular library with named subroutines (PASCAL, FORTRAN, etc.) $75

HARDWARE

Smith-Corona TP1 12 cps Daisywheel $895
IBM PC - 2 drive, 64K $3300
All IBM hardware/software available.

Intelliterm & DataMaster available now for TRS-80

FREE CATALOG

(and floppy disk SPEED-UP program listing)

☐ I have purchased _____IBM PC's _____K _____disks
☐ Mono ☐ Color ☐ Comm.
☐ Printer Type _____

Name _____

Company _____

Address _____

_____

City _____ State ___ Zip _____

Phone ( ) _____

Source #_____ Compuserve #_____

Dealer inquiries invited; reply on letterhead. ST 5/82

offers two time-saving products: *Time Scheduler/Organizer* allows you to not only schedule appointments, but also to automatically schedule any recurring activity or scan the calendar for available time blocks. Requires printer and eighty-column display. $99.

Δ *Magazine/Book Reference* is a reference program that permits recall based on publication, author, primary or secondary topic, or key word. Requires printer and eighty-column display. $69.

Δ *Micro/Terminal* is a communications program that allows information exchange with other microcomputers and with mainframes. Also compatible with the Apple II and III. A product of **Microcom** (1400A Providence Highway, Norwood, MA 02062; 617-762-9310). $99.95.

Δ **Architecture Technology** (Box 24344, Minneapolis, MN 55424; 612-935-2035) has scheduled the seminar "Selecting a Local Network" in the following cities: Boston (August 12-13), Minneapolis (August 19-20), San Francisco (August 25-26), and Washington, D.C. (September 9-10). The seminar will be conducted by Dr. Kenneth J. Thurber and Dr. Harvey A. Freeman.

Δ **Hewlett-Packard** (16399 West Bernardo Drive, San Diego, CA 92127; 714-487-4100) has a new plotter, the HP7585A, which is larger, faster, and has better resolution than the HP7470A plotter (Marketalk Reviews, July 1982). The new plotter can handle up to eight colors. $22,750.

Δ **Davong Systems** (1061 Terra Bella Avenue, Mountain View, CA 94043; 415-965-7130) offers a 12M Winchester disk system that can be installed in place of a floppy drive. $2,695.

Δ *Some Common Basic Programs*, from **Osborne/McGraw-Hill** (630 Bancroft Way, Berkeley, CA 94710; 415-548-2805), has been published in an edition for the Personal Computer. The book contains seventy-six programs for business and home. $14.99.

Δ **Interface** (20932 Cantara Street, Canoga Park, CA 91304; 213-341-7914) has a line of disk drives: single-sided internal, $470; single-sided external, $570; double-sided internal, $570; and double-sided external, $670.

Δ **L.J. Kutten** (201 South Central Avenue, Box 16185, Saint Louis, MO 63105; 314-721-6644) publishes *Consumer Protection for the Microcomputer Owner*, a book that discusses such things as what to do about defective merchandise and when a sale is final. $15.

Δ New from **Lifeboat Associates** (1651 Third Avenue, New York, NY 10028; 212-860-0300) is *Emulator/86*, which allows you to run CP/M-86 programs in PC-DOS. $75. Δ Also, *EM80/86*, an emulator for CP/M-86. $200. Δ *Ascom* is an asynchronous communications program designed for file transfers or interaction with time-sharing services. $175. Δ *Pmate* is a text editor based on single keystrokes that can be changed to correspond to a previously familiar editor. Requires two disk drives. $225. Δ *Lattice C Compiler*, a compiler for the language C, is compatible with Unix Version 7. Requires two disk drives. $500. Δ *UT86* is a system utility that provides formatted and sorted directories, interactive copy routines, and formatted file printouts. $180.

Δ The latest from **VisiCorp** (2895 Zanker Road, San Jose, CA 95134; 408-946-9000) is *VisiFile*, a menu-driven database program that can be used for anything from mailing lists to inventory management and allows for password protection and up to eighteen types of indexing. Requires two disk drives. $300.

Δ **Hurricane Labs** (5149 Moorepark Avenue, Cupertino, CA 95129; 408-257-8676) has two boards for the pc, a prototype board ($55, $50 kit) and an extender board, which can isolate the I/O channels through dip switches ($120, $80 kit).

Δ **MicroMatch** (10343 Commerce Avenue, Tujunga, CA 91042;

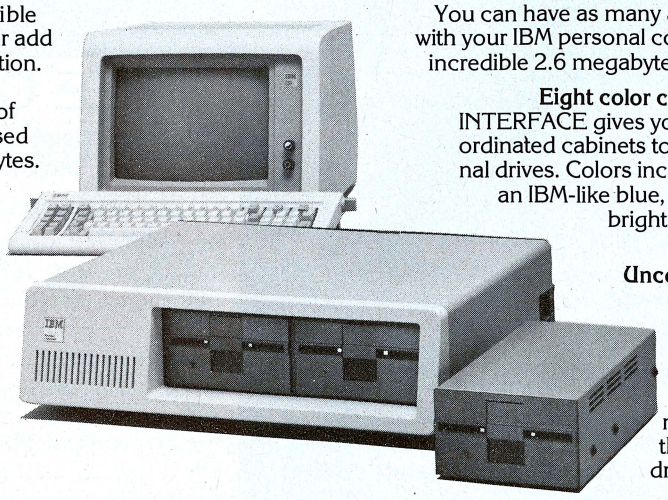
Introducing low cost, IBM compatible disk drives

Up to 2.6 megabytes of IBM formatted storage

Low cost internal/external drives
Plug the INTERFACE IBM compatible disk drives into your IBM cabinet or add on to your existing drive configuration. INTERFACE drives can be used externally or internally, in a variety of configurations. To give you increased storage capacity up to 2.6 megabytes.

Internal Drives
- Single sided with 168K bytes IBM format $215
- Double sided with 336K bytes IBM format $385
- Double sided with 650K bytes IBM format $570
- Two double sided drives with 1.3M bytes IBM format $1140

External Drives: (Includes drive, power supply, cabinet and cable)
- Single sided with 168K bytes IBM format $315 (Maximum two drives externally)
- Double sided with 336K bytes IBM format $485 (Maximum two drives externally)
- Double sided with 650K bytes IBM format $670 (Maximum two drives externally)



Up to four drives
You can have as many as four drives operating with your IBM personal computer. Giving you an incredible 2.6 megabytes of formatted storage.

Eight color cabinets to choose from
INTERFACE gives you a choice of color coordinated cabinets to choose from for external drives. Colors include standard off-white, an IBM-like blue, lime green, dark green, bright orange, brilliant yellow, black, and chrome.

Unconditional warranty and service
90-day unconditional warranty plus service center for out-of-warranty service. For more information about the IBM compatible disk drives, call or write today:

I² INTERFACE, INC.
7630 Alabama Ave., Unit 3
Canoga Park, CA 91304
(213) 341-7914

IBM is a registered trademark of IBM Corporation.
MasterCard, VISA and COD orders accepted.
Dealer and quantity discounts available upon request.

213-353-5929) has a line of RAM expansion boards with a five-year warranty, ranging from 64K ($400) to 256K ($730).

Δ **The Bench Collection** (1387-D Cass Road, Santa Rosa, CA 95401; 707-526-1074) manufactures Electroniture—solid oak furniture for personal computers. Combination cabinet-desk includes a built-in monitor cabinet with nonglare glass screen, document holder, software storage space, and room for a small printer. Matching printer stand for larger printer is also available. Cabinet-desk, $895; printer stand, $225; optional wire paper bale, $25.

Δ *EasyPlanner*, a new spreadsheet simulator from **Information Unlimited Software** (2401 Marinship Way, Sausalito, CA 94965; 415-331-6700), allows programmed spreadsheets. The concept is designed to make data entry by unskilled office personnel automatic and uniform. The user can organize a sequence of events—from loading a template to printing it—and provide prompts to assist those who enter the data. Programming also eases consolidation of spreadsheets. Two disk drives required. $195.

Δ **The Software House** (Box 112, Citrus Heights, CA 95610; 916-723-9111) offers a set of five utility programs. *Files* displays a disk directory in three sorted columns; all files on disk appear on a single screen, in either forty or eighty columns. *List* lets you speak to your serial printer with the print-screen key or the Basic *Lprint* statement. *Dump* provides screen display of the contents of files, each line showing eight bytes of data in ASCII and hexadecimal. With *Hex* you can step through a disk file sector by sector and examine its contents—in hexadecimal and ASCII. And, in the same format, *Sector* displays the contents of any sector on disk, including the directory and other sectors reserved by the operating system. Any one program, $20; additional programs, $5 apiece ($40 for the set).

Δ The **Supersoft** (Box 1628, Champaign, IL 61820; 217-359-2112) *C Compiler* is now available for the pc in separate versions for CP/M-86 and PC-DOS. Each, $500.

Δ **Applied Business Computer** (1885 North Main Street, Orange, CA 92665; 714-921-0130) offers a memory board expandable to 512K. Prices from $499 (64K) to $1,384. Δ Same company has an EPROM programmer that handles 2764, 2564, 68764, 2732, 2732A, 2716, 2516, 2758, and 2508 EPROMs. $295.

Δ **JC Computer Specialists** (Box 3465, Federal Way, WA 98003; 206-839-8268) offers two memory joggers—easel-mounted legal-sized reference cards. One is for PC-DOS, the other for Basic. Each, $10.

Δ *PMS-II* is a critical path network analyzer and project management system. The program computes critical paths and floats with early and late start and finish, and allows a thousand activities per network (more if you have more than 64K). It provides Gantt charts, activity diagrams, funding schedules, and more. From **Aha** (1475 Rodriguez Street, Santa Cruz, CA 95062; 408-475-8705). Requires CP/M-86 and CBasic-86. $1,295.

Δ TR1M stands for "total RAM one megabyte," a memory expansion card designed for use with the 256K RAM chips to be released late this year. It will also work with most 128K and 64K chips. From **Legend Industries** (2220 Scott Lake Road, Pontiac, MI 48054; 313-674-0953). Price to be announced.

Δ **Innovative Computer Products** (18360 Oxnard Street, Tarzana, CA 91356; 213-996-4911) sells cleaning kits: a head cleaning kit for disk drives, $29.95; a computer maintenance kit, $39.95; a peripheral maintenance kit, $24.95; a display cleaning kit, $12.95; and an anti-static kit, $59.95.

Δ *PC Perspectives Newsletter*, a monthly journal for the professional pc user, is slated to start publication in September. The journal will concentrate on the application of software and hardware in office and factory environments. Yearly subscriptions from **Architecture Technology Corporation** (Box 24344, Minneapolis, MN 55424; 612-935-2035). $195 (U.S.), $245 (foreign). ▲

PUT YOUR IBM PC TO WORK WITH

THE DATA REPORTER

With the Data Reporter you receive an advanced user-friendly data management tool that delivers more power and flexibility than any other database system available for the IBM Personal Computer. This information management package consists of a unique combination of high speed machine language routines and BASIC programs. The modified "Program Generator" set-up program easily customizes the package to handle your data your way. Powerful search and sort routines let you manipulate your files to your requirements. The revolutionary "Soundex" search routine is included so you do not have to know the exact spelling of the data you are trying to access. Special features include fast data entry and update, calculations, data analysis, label maker, merge, data and program file back-up, reformat, global editing, and more. The report generator has many word processing features which allow you to print reports, form letters, and data in a variety of ways, with or without calculations. The Data Reporter uses single keystroke, menu driven commands with easy to follow prompting; non-programmers will have no problems using this powerful package. Sophisticated error avoidance and recovery techniques prevent undesired results. The BASIC language programs can easily be modified for those desiring special applications, while the machine language routines provide unexcelled speed. Use The Data Reporter for any of your data handling needs.

The program innovators at Synergistic Software will bring you this and other helpful, as well as entertaining, IBM PC software.

Send for detailed information to:

Synergistic Software

830 N. Riverside Dr., Suite 201
Renton, WA 98055
206-226-3216

Exec Tecmar

The Expansion Team

Martin and Carolyn Alpert launched Tecmar from the living room of their graduate student apartment. Dave Wertman, right, subsequently came on board as vice president of marketing.

by David Fitzgerald and Jim Salmons

"It was a high atmosphere. Food was being brought in. We were open twenty-four hours a day, seven days a week. A large crew was here day and night. Everybody's sights were set on one goal—making the Comdex deadline."

The speaker is Martin Alpert, the company Tecmar, Incorporated. Alpert is describing the frenzy of activity that preceded last November's Comdex show in Las Vegas, where Tecmar performed what many industry watchers considered an unaccomplishable feat—the release of twenty full-blown working peripheral products for the IBM Personal Computer.

With just six weeks between IBM's first retail sale of the pc and the opening of the leading microcomputer industry trade show, expectations were that companies wishing to enter the pc peripheral marketplace would release specifications of their full IBM-compatible product lines, buttressing these announcements with demonstrations of prototype samples. The more aggressive companies might, at most, have been expected to release one or two fully developed products.

Tecmar's stunning twenty covered a broad spectrum of scientific, engineering, and industrial applications and also provided for a full range of general-purpose system expansion. It was a feat that lowered competitors' jaws in disbelief and raised approving eyebrows at IBM.

How did they do it? Neither by venture capital nor by corporate borrowing, but by making a habit of hard work—right from the company's cottage-industry, grad-student apartment beginnings.

Martin and Carolyn Alpert met in 1971 as undergraduates at Case Western University in Cleveland, Ohio. They managed to weave a marriage into the fabric of two equally busy graduate student schedules. Martin pursued the master's degree in systems control engineering while Carolyn sought an advanced degree in biology.

Martin, whom associates today describe as a "technology freak," began a project to create a mathematical model of the human lung. The intellectual stimulus of that work enhanced his growing skills as a designer of computer systems. In using his newly acquired technical knowledge, he opened an entirely new field to his inquiring mind.

Is There a Doctor in the House? Motivated by a desire to create high-technology medical products, Martin entered Case Western medical school. Carolyn meanwhile studied part-time while working two jobs—teaching college and doing biological research.

Most couples would have found this regimen fulfilling, but the exceptional Alperts had time on their hands. His imagination fueled by his earlier work in lung modeling, Martin began development on a revolutionary pulmonary diagnostic tool that, by prevailing professional opinion, had no chance of success.

This instrument was designed to replace a crude and disconcerting procedure that measured the elasticity, or compliance, of the lung. This procedure involved placing a balloon-tipped tube into the patient's esophagus, by way of the nose. After the patient was placed in an airtight box, the balloon was inflated and the patient was asked to breathe in a set pattern. All this misery was required for the technician to obtain a single measurement.

By 1979 Tecmar would be able to demonstrate a space-age instrument that could accurately measure lung compliance every forty-thousandth of a second, without requiring any activity on the part of the patient. The instrument would be wheeled to bedside, and the patient would breathe normally into a comfortable mask. The opinions of Alpert's medical peers would do an about-face.

Congratulations—It's a Corporation. The means by which this revolutionary product came into being would themselves constitute a significant accomplishment—the birth of a healthy high-tech corporation. During Tecmar's corporate gestation, Martin was busy designing the instrument while Carolyn performed support work—purchasing components, building and testing prototypes, and remaining gainfully employed.

The Alperts' part-time project quickly became a full-scale business, leading to Tecmar's incorporation in 1974. Carolyn altered her career plans and transferred her science credits toward an MBA.

The routine of hard work and study continued to stimulate product-development activity in the couple's living room.

The first additions to the Tecmar executive team arrived while the company was still in its living-room phase. Kenneth Stern, now vice president of engineering, was finishing a doctorate in biomedical engineering at Case Western when he heard about the Alperts. Drawn by their entrepreneurial enthusiasm, Stern volunteered his free time.

Before moving to Cleveland to complete his studies, Stern had worked in hardware at Bell Laboratories. His training and experience complemented and expanded the Alperts' skills.

As a means to the long-range end—the development of sophisticated medical equipment—Tecmar began concentrating on elementary data acquisition and processing devices crucial to laboratory and industrial process control applications. With a move into its first separate business location and the expansion of its administrative, production, and technical staffs, Tecmar released its first line of products for the scientific and industrial markets.

The initial Tecmar line grew from a limited selection of digital-to-analog and analog-to-digital converters into a broad range of laboratory hardware and software products, culminating in the TecLab and Tec-86 laboratory microcomputer systems.

TecLab is a Z-80–based system that bundles data acquisition, storage, analysis, and display devices. Running under CP/M, the system offers Microsoft Fortran and Basic as well as the TecLab software support package.

TecLab's product companion, the Tec-86, was an early entry into the sixteen-bit 8086 microcomputer market. Combining a similar assortment of laboratory control devices, Tec-86 provides the increased mathematical precision and speed demanded by scientific and industrial users. Tec-86 runs under CP/M-86 and offers Basic-86 as well as an expanded TecLab software package capable of handling its multiple sixteen-bit input/output ports.

What's Past Is Prologue. The experience and knowledge gained during the development of Tec-86 would ultimately provide a foundation for Tecmar's assault on the IBM Personal Computer.

Tecmar was inching its way toward increasingly sophisticated and specialized medical equipment. But the cost of such development was staggering. Determined to assist this effort with retained earnings from sales of spin-off products, Tecmar found itself drawn to the growing base of Apple, Commodore, and Tandy microcomputers.

Management concluded that its strategic focus on highly technical markets was too limited—and too limiting, preventing the company from taking full advantage of the microcomputer revolution of the late seventies and early eighties. An entirely new distribution channel was developing within the network of independent microcomputer retailers. A more broadly based marketing strategy would be needed to usher in the next phase of Tecmar's corporate development.

To design and execute that strategy, the executive triad brought Dave Wertman aboard as vice president of marketing. To maintain the corporate accounts within existing specialized markets, William Niro was added to the staff as director of sales and distribution. And then, in order to take advantage of the expanding microcomputer market, the company formed a cooperative effort between marketing and product design to develop a line of peripheral products that would interface with S-100 buses as well as with the Apple.

Niro Fiddles While Tecmar Earns. Niro introduced these new products through Tecmar's existing channels, while Wertman focused on the creation of new channels, such as retail sales through the Computerland network. "It was a new ball game for us," says Wertman, "one in which we were determined to succeed."

Martin recalls the frantic team-building days of the pc product development effort. "We saw the opportunity early, as soon as IBM announced their Personal Computer. Of course, our own 8086 product line gave us insight into what to expect. We spec'ed out the product the best we could through industry rumors."

The company began software development of the user interfaces while waiting to get their hands on a machine to begin hardware interfacing. They flew to Chicago to pick up the first two pcs sold retail. After bringing the new machines back to the shop, they used logic analyzers to start de-engineering the pcs.

No Magic Bus. "Fortunately," explains Alpert, "it was a straightforward bus. There were a couple of tricks to it that took longer, but we figured it out quite easily. Then it was a matter of how fast we could attach the bus to the user interfaces we had previously developed."

Tecmar's sophisticated production graphics section was moving quickly behind the engineering design team. They had to take the system designs of the logic diagrams and wiring schematics down to the circuit templates required for board printing production.

"To really get a jump on the market," says Alpert, "we realized the need to increase our production capabilities radically, and even then we were bursting at the seams."

So they subcontracted production all over Cleveland. "We set up a metal fabricating plant that was at our disposal. And we had an auxiliary printed circuit board house geared up so as soon as the graphics people finished the production templates we started cranking out boards. It went like clockwork."

According to Alpert, the work was extremely organized. But he doesn't want to make it sound too easy.

"It was a tremendous amount of effort," Alpert admits, citing employee dedication throughout the organization as the key to making the Comdex deadline. "People were putting in thirty-six-hour shifts. Everyone had their individual goals, every day. If they met their goals early, they didn't go home. They helped those who were behind, because everything had to be done serially. If somebody got ahead on wire wrapping, they'd help out with the taping of a board."

With its initial pc product line in place and system manufacturers banging on the doors, Maw Lee was added to the team as general manager, a position of critical support to Carolyn's operational management activities.

Inside Job. Commenting on his initial goal—to remove the kinks from the production pipeline—Lee observes, "Tecmar's ability to do things quickly involves our having control over just about every aspect of development and production." The company has its own graphics, photography, and typesetting. Their two printing presses are almost constantly running. They have a machine shop on the premises to support prototype development and production tooling, with a broad range of production capabilities—including wave soldering, chip burning, and component assembly and testing.

Although innovative ideas for new products abound at Tecmar, the development of new products is controlled. Every new product must meet at least one of two strict criteria. Says Alpert: "Either the product must be unique to the marketplace or it must add value to an existing product, in response to a demonstrated consumer need. Most of our products do both."

At the June 1982 Comdex in Atlantic City, New Jersey, Tecmar released three new products for the pc. The first is an eight-inch floppy disk drive. An add-in option to Tecmar's expansion chassis, the eight-inch drive is a response to the needs of users running available CP/M-86 applications.

The second product, a multifunction card, answers a variety of needs as well as responding to a need in the EOM software publishing market. The multifunction card can be configured with 64, 128, 192, or 256K RAM, provides a battery-backed time and date function, and includes both serial and parallel ports for interfacing printers and other external devices.

Secret Agent. The board also includes an encryption decoder that provides a valuable security feature for the business software industry. Besides a lock to ensure the "one program to one machine" concept inherent in most software licensing agreements, the decoder allows the incorporation of multiple security codes into a publisher's program, as a means of governing access to data.

The third product released at Comdex demonstrates Tecmar's expanded consumer focus and marketing strategy. In recognition of the need to minimize the desktop space dedicated to a pc installation, Tecmar has designed and manufactured a modular cabinet that fits under most conventional desks. Called the PC-Rover, it mounts the pc chassis vertically in a self-contained unit that can be rolled from one desk to another, wherever the computer is needed.

"Though we expect a good deal of excitement about these new releases," says Wertman, "we're saving our most exciting new product for a debut of its own. It's a product that we believe will push the personal computer a development generation ahead. Just thinking about it gets me excited."

Adds Alpert: "We have a 68000 processor board that plugs into the Personal Computer. It will allow you to run 8088 and 68000 software concurrently. You'll be able to address at very high speeds a full sixteen megabytes of memory with the 68000, while simultaneously running software on the 8088."

For operating-system support of the 68000 board, Tecmar has made arrangements with the Mark Williams company to offer the Coherent operating system—described by Alpert as "a very nice Unix look-alike." A number of high-level languages will be available for the board as well.

Coming Attractions. Slated to follow the August release of this expansion board is an additional line of specially designed memory add-ons, multifunction cards, and interfacing devices. These general-purpose system expansion products will round out the PC-Mate family. Judging by past performance, Tecmar will probably deliver these additional devices before the end of the year.

Currently, Tecmar has a commanding lead in the pc peripheral market. Their present line ranges from dust covers to protect the pc, to printers and an expansion chassis.

The chassis serves a dual purpose. Its primary function is the housing of seven extra peripheral slots, but it may also carry a five or ten megabyte Winchester drive.

For the electrical engineer, Tecmar's extender cards and prototyping boards help speed development projects. Laboratory and industrial technicians may interface their analog or digital signals through Tecmar's A/D and D/A converters as well as interfacing any IEEE-488–compatible device directly to their Personal Computers.

The business user may save money by sharing one printer among as many as four Personal Computers, by means of the Shared Device Controller. Word processing and report generation may be made more efficient with the RAMSpooler utility. Time and date document fingerprinting may be performed with the Time Master card. Unbelievably fast communications, at up to 230K baud, are supported by Scribe Master, allowing a variety of networking configurations among pcs and high-speed external devices.

It Sees. It Moves. It Talks. For pioneers in industry, a Stepper Motor Controller allows computer-aided design and manufacturing experimentation. A video digitizer provides pattern recognition processing, opening avenues to robotics research. For other kinds of pioneers, a variety of applications may be found for Speech Master, which offers twin synthesizers for the ultimate flexibility in computer speech.

What else lies ahead for Tecmar? "Growth," says Carolyn. "A move to a new location is crucial. We're pushing the limits here at around eighty employees. We've located an ideal space that will allow us to expand up to 200,000 square feet."

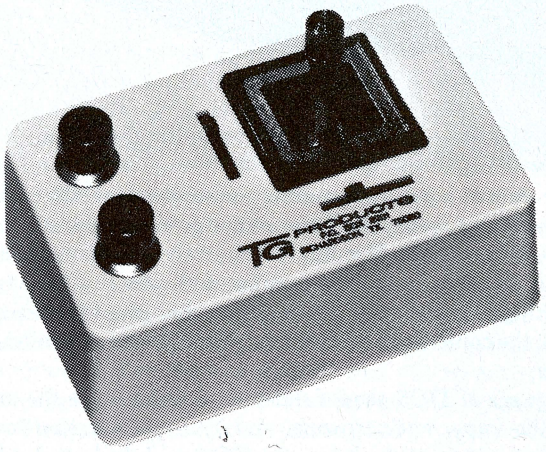
For Martin, the continuation of long-range development of highly sophisticated medical equipment remains a vital motivation.

Stern is excited about another long-range project. The company is in the late development and early prototyping stage of a voice recognition unit. "With a speech synthesizer talking on one computer and the voice recognition unit listening on another," says Stern, "we can actually have one computer tell the other what to do—in plain English. A lot more programming and natural language analysis will be necessary before this idea can really work, but it holds incredible potential."

The depth of this research focus and the "can do" attitude of its management reflect Tecmar's motto: "The difference between fantasy and reality is tomorrow." For Tecmar, in fact, that gap may be no greater than from here to the next Comdex. ▲

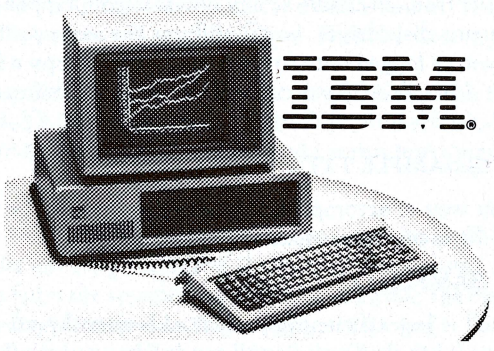
NEW . . . . ANNOUNCING . . . . NEW

JOYSTICK



- Large buttons
- Molded cable
- Highest quality components
- Completely linear operation
- Exclusive trim adjustments

$64.95



Designed expressly for the IBM® Personal Computer.

Not just a quick adaptation.

Custom made components required over four months of design effort.

Available at your dealer or order direct.

Allow 2 weeks for shipment.
All mail orders add $2.00 for postage and insurance.
Dealer inquiries invited.

Texas Residents add 5% tax.

IBM is a Reg. TM of International Business Machines Corporation.

TG PRODUCTS
1104 SUMMIT AVE., SUITE 106
PLANO, TEXAS 75074
(214) 424-8568

SYSTEM NOTEBOOK

by Alan Boyd

To date we've seen how to use some of the most basic DOS commands. This month we'll continue by examining more of these commands. At this early stage, we'll still restrict ourselves to the types of commands that all users need to be familiar with—the utility commands for making backup copies, erasing files, changing file names, and displaying the contents of a file on the video screen or printing them on paper.

In the language of DOS these fundamental utilities are the internal commands—*copy, erase, rename,* and *type.* As we saw last month, these commands are built into the DOS module that's loaded into memory whenever the system is booted. You can invoke these commands at any time, simply by typing the command name at the keyboard, along with whatever arguments may be required. You don't have to have the DOS disk in a drive to invoke an internal command.

Other DOS commands, known as external commands, are resident on the master disk and are read into memory each time you invoke them. One consequence of this division of the command structure into internal and external commands is that the internal commands are available as soon as the computer is booted with any user-created master disk, regardless of whether the other DOS command files have been transferred to the disk.

Last month's column explained how to format a blank disk, using the *format* and *format/s* commands, and how to copy the contents of one disk to another by means of the *diskcopy* command. *Diskcopy* is an external command and will probably be used less frequently than the internal command *copy*. *Copy* is the first command we'll examine in depth this month.

Packing Protection. The *copy* command is an internal DOS command used to make duplicate copies of individual files and transfer them from one disk to another—or even to the same disk. During the copy process the name of the copied file may be changed. The primary use of the *copy* command is to make backup copies of individual files. Backing up important data files is highly recommended and is the single most important deterrent against accidental erasure of valuable data.

The *copy* command may take various syntactical forms, depending on your disk drive configuration and how you intend to use the command. As we pointed out last month, you log onto a disk drive by typing the letter name of the drive followed by a colon. For example, to log onto the B drive, you type

>B:

and hit the enter key (you don't type the greater-than symbol; that's a prompt from DOS). Unless stated otherwise, each of the following examples will assume that you're logged onto drive A.

The first form of the *copy* command looks like this:

COPY filespec

where *filespec* takes the form of

D:filename.ext

To illustrate, let's assume you're logged onto drive A and that you want to copy a file named Sample.txt from drive B onto the disk in drive A. To accomplish this you would type

COPY B:SAMPLE.TXT

This would have the effect of reading the file Sample.txt from the disk in drive B into memory and then writing it onto the disk in drive A. If the file happened to be too big to be read into the memory available on your system, then your computer would read as much of the file as it could fit into memory at one time, and then write that much out to the target disk. It would then repeat those steps until the entire file had been transferred.

The second form of the *copy* command can be illustrated by an example that reverses this process—it copies a file from the logged drive (A) to the disk in drive B. The syntax of the *copy* command required to accomplish this would be

COPY filename[.ext] d:

The *d:* in this case is the drive designation of the target drive, the drive containing the disk onto which the file is to be copied. For example, let's assume you're logged onto drive A and you want to copy a file named Sample.txt from drive A to drive B. To do this you'd type

COPY SAMPLE.TXT B:

By combining these two examples, you can see that you can copy any file from any drive to any other. If you happened to have more than two disk drives, you'd still use the same syntax. If, for example, you're logged onto drive A and want to copy a file named Sample.txt from drive B to drive C, then your command format would be

COPY B:SAMPLE.TXT C:

Another way to accomplish the same result would be to log onto drive B first, and then issue the command

COPY SAMPLE.TXT C:

This method is less efficient because it adds more keystrokes to the process, but it has the same overall result. How you do it is entirely up to you.

The next form of the *copy* command would be used in a situation where a file must be transferred from one drive to another and the name of the file must be changed. The full syntactical format for this usage of the command could be described as

COPY filespec filespec

In this case the first *filespec* is the name and drive designation of the file that is being copied from—the source file—and the second *filespec* is the name and drive designation of the file that is to be created—the target file. To illustrate, let's assume you're logged onto drive A and that you want to copy the file Sample.txt from drive A

to drive B; at the same time, you want to change the name of the new file to Backup.txt. The command format would be

COPY SAMPLE.TXT B:BACKUP.TXT

Suppose further that, while still logged onto A, you want to copy the file Example.txt from drive B to drive C and again you want to change its name. In this case you'd issue the command

COPY B:EXAMPLE.TXT C:BACKUP.TXT

In the first example the drive designation for the source drive, *A:*, was omitted because it was the currently logged drive and didn't need to be entered.

If you're ever uncertain about whether a drive designation needs to be included in a command or not, it does no harm to include it; redundancy is fine, even if it requires a few extra keystrokes to achieve.

The *copy* command can also be used to create a backup of a file on the same disk as the source file. You do this by specifying different file names and/or extensions and by either omitting the drive designation or using the same drive designation for both the source and target files.

For example, to make a backup copy of the file Sample.txt on drive A and call it Backup.txt, you'd issue the command

COPY SAMPLE.TXT BACKUP.TXT

If the file was on drive B, the command would be

COPY B:SAMPLE.TXT B:BACKUP.TXT

It's easy to see that if a backup copy of a file is to be created on the same disk as the source file, then the newly created file must be given a file name or extension that differs from the file name or extension of the source file. Otherwise DOS wouldn't accept the command, since it would lead to the intolerable situation of having two files on the same disk with the same name. If you happen to make this mistake, DOS will present you with the error message:

File cannot be copied onto itself
0 File(s) copied

Another interesting feature of the *copy* command is its acceptance of commands incorporating the wild card characters—the asterisk and question mark. As you may remember, these characters may be used to replace other characters in file names. The wild card feature allows you to copy several files by issuing a single command.

The simplest example of the use of wild cards in a *copy* command would be the case where all the files on a source disk are to be copied to a target disk. The command for this operation would be

COPY *.* B:

This use of the *copy* command would have substantially the same effect as the *diskcopy* command discussed in last month's column; everything from the source disk would be copied to the disk in drive B, and all the file names would remain unchanged.

Similarly, if you wanted to copy all the files with a given extension from the logged drive to drive B, you could issue the command

COPY *.EXT B:

There are some other ways in which the wild cards can be used to simplify commands. For example, if you want to copy a file called Sample.txt from the logged drive to drive B, and at the same time you want to rename it to Sample.bak, then, rather than typing

COPY SAMPLE.TXT B:SAMPLE.BAK

you may use the shorter and simpler command

COPY SAMPLE.TXT B:*.BAK

Similarly, if you want to copy all the files with the extension *.txt* and give each of the copied files the extension *.bak*, you could accomplish that with the command

COPY *.TXT B:*.BAK

Before we leave the subject of the *copy* command, a short discussion of the differences between the internal command *copy *.** and the external command *diskcopy* is in order. Although both of these commands appear to perform the same task, there are definite differences in both the method and the results.

The *diskcopy* command makes an exact duplicate of the source disk. Therefore, if the source disk is a master (one that can be booted directly), the copy will also be a master.

Such is not the case with the *copy* command, even if *copy* is used globally, as in *copy *.* b:*. *Copy* will only transfer nonsystem files and therefore will not create a master disk.

There's one situation in which it makes more sense to duplicate an entire disk with *copy *.** than with *diskcopy*. That's in the case where file access is being hampered by nonlinear distribution of space on a disk. This can happen when a disk is used often, or when files are erased from a disk and the disk is subsequently used to store other files. To understand what this means it's necessary to know a little about how DOS allocates disk space for files.

Finder's Keepers. When you ask DOS to store a file, it allocates sectors on a first-come, first-served basis. That is, whenever a file requires more space from DOS, the operating system gives it the next available sector as shown in the File Allocation Table.

This first-come, first-served approach means that disk files may not be distributed sequentially on a disk, but may be scattered around the disk in a seemingly random manner. The likelihood that a file will be stored in noncontiguous disk sectors increases the more you erase files from disk. When you invoke the *erase* command, you

PC/FORTH

Complete FORTH program development systems for the IBM® Personal Computer. Packages include interpreter/compiler with virtual memory management, line editor, custom screen editor, assembler, decompiler, utilities, file and record access modules, and many demonstration programs. 100 page user manual $100.00

Floating point arithmetic extensions and cross compiler available at additional cost.

Ready to run under PC-DOS or CP/M-86®. Standalone version available soon. 64 kbytes RAM and two disk drives recommended. (IBM is a registered trademark of International Business Machines Corp. CP/M is a registered trademark of Digital Research, Inc.)

Laboratory Microsystems
4147 Beethoven Street
Los Angeles, CA 90066
(213) 306-7412

free previously allocated sectors; the next file that gets written to that disk will be allocated the first available sector, which will probably be one that was freed during the erasure, and the result of this process is that your files will become more and more disjunct in their physical locations on the disk.

When you copy a disk with the *diskcopy* command, you duplicate on the target disk whatever randomness of organization may exist on the source. When you back up by means of *copy *.**, on the other hand, the operating system writes files in contiguous blocks on the target disk. So one important function of the *copy* command, as distinguished from *diskcopy*, is to "straighten out" the file allocation for a set of files.

You might want to use this technique to improve file access time on disks where DOS seems to take longer than usual to store or retrieve data. Disks that store an index of some kind, where the index points to data contained in other disk files, are especially suitable candidates for straightening out by means of *copy *.**. Indices that are stored in contiguous sectors on a disk may perform considerably better than their randomly written counterparts.

There are other, more advanced, uses for the *copy* command, but since they involve some of the more complex features of DOS, we'll defer them until we have established a good base of understanding for the universally applicable DOS commands.

One final note about *copy* for those who are used to CP/M. *Copy* does substantially the same things as the *pip* command in CP/M. The major difference is the order in which arguments are stated. *Pip* requires you to indicate the target disk before the source disk; DOS takes the more intuitively obvious approach of putting the source disk first.

Erase. The *erase* command was introduced as an example in last month's column. It is an internal DOS command whose purpose is to remove a specified file or group of files from a disk. The syntax is

ERASE filespec

where *filespec* follows the general rules for file specifications. For example, to remove the file Example.txt from the disk in drive B, you would enter the command

ERASE B:EXAMPLE.TXT

As with the *copy* command, wild card characters may be used in a file specification to simplify the erasure of several files at once. For example, to remove all files with the extension .bak, you could issue the command

ERASE *.BAK

To erase all files on a disk (a particularly drastic measure), the single command

ERASE *.*

would suffice. Treat this form of the *erase* command with kid gloves. Before you erase all the files from a disk, make sure that's really what you want to do—by looking at the directory (with the *dir* command). The use of *erase *.** is tantamount to reformatting the disk.

The *erase* command cannot be used to remove the system files from a disk. That is, it will not operate on the files Ibmbio.com, Ibmdos.com, or Badtrack. If you want it to, however, it will remove the file Command.com.

If you elect to remove Command.com from a disk, you'll need to replace it with another file of the same name, or else there will be no way for you to communicate with DOS via the keyboard. That's because Command.com contains the Command Line Interpreter, the link between the visual command line and the operating system. This is not something to be done by a beginner, since a user-supplied Command.com must perform many nontrivial functions. If the Command.com file is erased from a disk and you try to boot with that disk, then the system will return an error message and will not be able to function.

When you invoke the *erase* command, it doesn't actually remove files from the disk. What it does is mark the files as erased and modify the File Allocation Table to indicate the newly available disk sectors. DOS will then allocate these sectors to other files as the need arises.

So the erased data is never actually eradicated from a disk until it is overwritten. It is, in fact, possible to reconstruct an erased file by hand, providing that DOS has not written anything new to the sectors involved since the file was erased. There are, as of this date, at least two utility programs on the market to accomplish this feat, and there will probably be others.

Erase performs in substantially the same way as the CP/M command *era;* the two extra characters in the DOS command make it a longer command to type and thereby perhaps reduce the chances of your typing it by accident.

Rename. The *rename* command is an internal DOS command used to change the name by which a file is stored on disk. The general syntax is

RENAME filespec filename

where *filespec* refers to the current name and drive designation of the source file and *filename* refers to the new name. For example, to change the name of the file Oldfile.txt on drive B to Newfile.txt, the command would be

RENAME B:OLDFILE.TXT NEWFILE.TXT

The wild card characters ? and * may be used in conjunction with the *rename* command, simplifying the example just stated to

RENAME B:OLDFILE.TXT NEW????.*

or even to

RENAME B:OLD*.* NEW*.*

providing that Oldfile.txt was the only file on the disk in drive B whose name started with the characters *Old*.

A File by Any Other Name. If you give a drive designation to the new name, it will be ignored by DOS, because it's meaningless to rename a file on a drive other than the drive on which the file is located. *Rename* simply alters the contents of the disk directory, substituting the new name for the old. Obviously, the new name must obey all the rules for DOS names.

There is one situation in which, theoretically, the use of the *rename* command could lead to disaster. That's when the new name you give the file is the name of a file that already exists in the directory. This would lead to the unworkable situation of having two files of the same name on the same disk.

The consequence of this mistake would be that you would only be able to gain access to the file that was created first, since that would be the first file with the given name that DOS would find when it searched through the directory.

Luckily, however, PC-DOS protects against mishaps of this sort. If you try to rename a file by giving it the name of a file that already exists, DOS will return the error message:

Duplicate file name or
File not found

Speaks Your Language. Like *copy*, the *rename* command differs from its CP/M counterpart (*ren*) in the order in which it requires the arguments to be named. The CP/M equivalent of *rename b:Oldfile.txt Newfile.txt* would be *ren b:newfile.txt=oldfile.txt*.

DOS takes an approach to its syntax that is more familiar and more in tune with the way that we think and express ourselves in English. We think of the action as "rename oldfile newfile," which is syntactically the way that DOS requires the command to be is-

sued. CP/M, on the other hand, imposes a backward, right-to-left, syntactic structure. Those accustomed to CP/M need to be aware of these syntactic differences.

Type. The last DOS command we'll examine this month is *type*, an internal command whose function is to examine the contents of a file by printing it out—either on the video screen or to a printer. This command gives you a method of examining the contents of a file without modifying them. This method can only be used for data files that contain ASCII characters; it can't be used for program files or other binary files.

ASCII files are those that contain data. In such files, characters are stored as binary numbers by a coding system in which each of 256 numbers represents a specific character. Most of these characters are either letters, numbers, or punctuation symbols. Some, however, are what are called *control characters*; these are often used to issue instructions to peripheral devices. For example, a control-G sent to the printer tells the printer to ring its bell; sent to the video screen, it causes the computer to beep. Other control characters perform functions such as clearing the screen, issuing a line feed, changing the character set to be displayed on the printer, and so on.

Most data files will contain at most a few control characters for formatting—such things as carriage returns, line feeds, and tab characters. The *type* command is designed to work with files of this sort.

On the other hand, program files—those files that contain program instructions, DOS commands, or other code—store binary representations of codes that do not represent letters of the alphabet. Attempting to type such files will have totally unpredictable results, since a code that represents a simple program statement may, in a different context, be the code that tells a printer to switch from text mode to graphics—or something equally bizarre.

Fear of Filing. For safety's sake, don't attempt to type files that don't contain data. Do not attempt to type files with the extensions .com, .exe, .hex, .map, .rel, or .obj. You won't do any physical damage to your system by typing these files, but you could cause your computer to behave in an unpredictable fashion, particularly if you have peripheral equipment attached.

It's perfectly safe to type files with the extensions .bat, .asc, and .dat, or any file generated by an applications program that's known to contain data. You'll find the *type* command particularly useful for those files that are generated by word processors or other text editors.

The general format for the *type* command is

TYPE filespec

This is relatively straightforward, since there's only one file involved. *Type* requires that the filespec be unambiguous, so wild card characters cannot be used with this command.

Here's an example. To examine the contents of a file named Example.txt, located on the disk in drive B, you would enter the command

TYPE B:EXAMPLE.TXT

This would cause the contents of the file Example.txt to be displayed on the video screen. To stop the scrolling, you'd press the numlock key while holding down the control key. This key combination (called control-numlock) suspends the operation of the computer in any program, unless the programmer has deliberately made the control-numlock function inoperative. Once you've stopped the screen from scrolling, hitting any key will cause scrolling to resume.

Another special keystroke combination is control-printscreen. This one you invoke by holding down the control key and pressing the key marked *PrtSc*. Doing so causes all output to the video screen to be sent to the printer at the same time. By pressing this combination and then invoking the *type* command, you can dump the contents of a file to the printer as well as the screen.

All this can be accomplished by entering one line. For example, to dump the contents of the file Example.txt on drive B to the printer, the command

[CTRL-PRTSC]TYPE B:EXAMPLE.TXT

can be entered. When the operation is done, the printer can be disconnected by typing the control-printscreen combination a second time. This keystroke combination has the effect of toggling the printer on and off.

In dumping the contents of a file to the screen or the printer, the *type* command will recognize carriage returns and line feeds and respond appropriately. When it encounters a tab character, it will advance to the nearest even multiple of eight columns; that is, it provides built-in tab stops at columns 8, 16, 24, and so on, and it will jump to the next stop when it reads a tab character.

This may not be consistent with the tab stop arrangement under which the document was prepared, but there is no way for the *type* command to determine what that arrangement was. Except for this treatment of tabs, *type* will render your document exactly as it is stored on disk.

Next month we'll take a look at some of DOS's less commonly used external commands. ▲

What they're saying about **DISKLOOK**, the complete diskette displayer:

"Wow! That ***DISKLOOK*** is a real addition to DOS!" — Mike Todd, president, DC User's Group

"I wish I saw more programs this good; you should charge more for it!"— Jack Crone, editor, **PC Age**

"Thanks for writing ***DISKLOOK;*** we get good use from it at PC!" — Jim Edlin, founding editor, **PC** mag.

Two programs no IBM/PC should be without:

DISKLOOK — a full-featured diskette display program; shows hidden files, erased files, full sector displays, maps storage use and file location, and more. **$20**

UNERASE — recovers erased diskette files, even when partially over-written. Can you afford to be without ***UNERASE***? **$20**

More utilities for the advanced IBM/PC user:

FILEHIDE — hides and unhides files, protects files from erasure **$10**

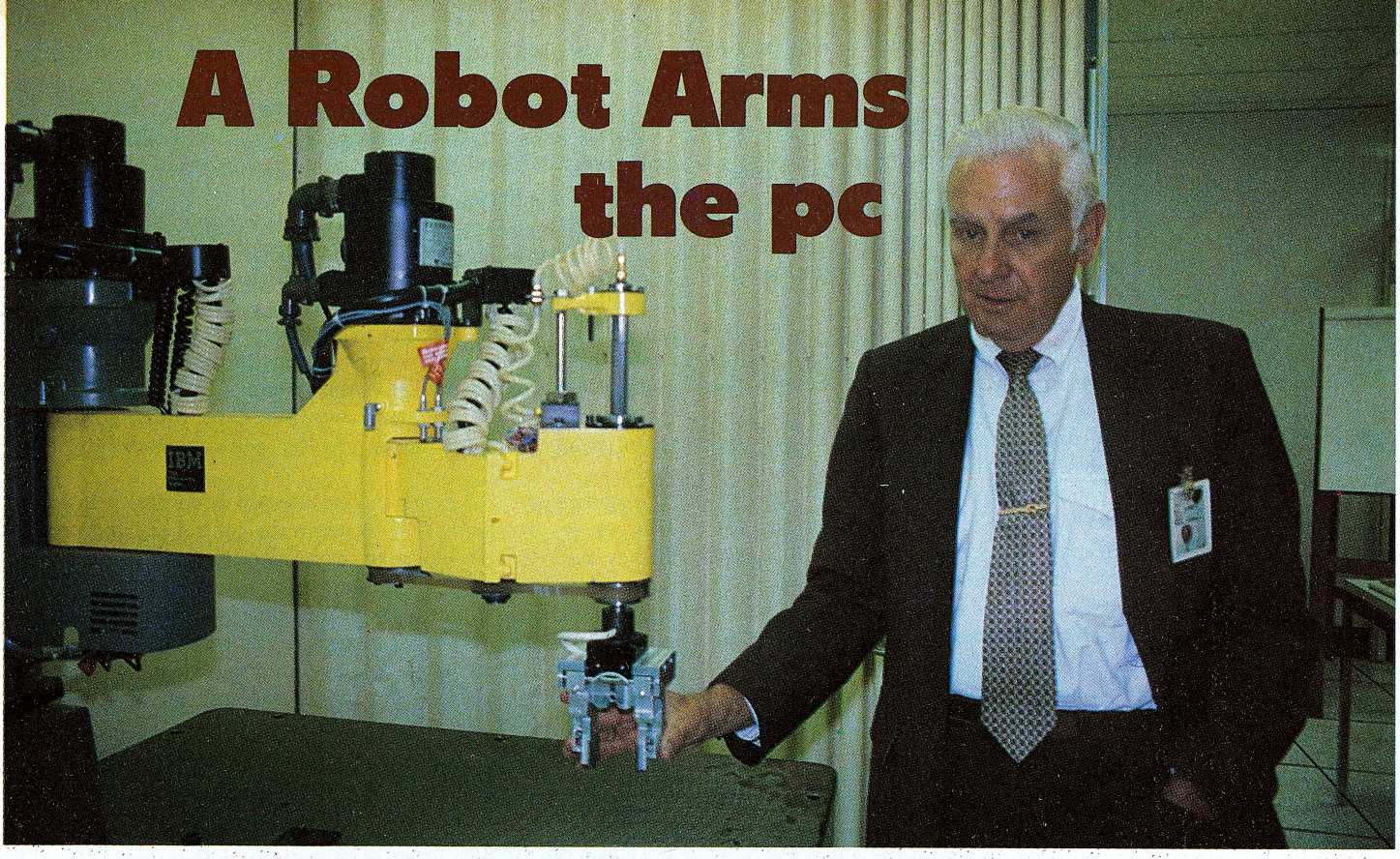
SECMOD — reads and easily modifies diskette sectors, in hex and ASCII; a patcher's dream **$20**

THE NORTON UTILITIES

For all orders, add $5 for diskette postage & handling; California 6% tax

Peter Norton 1716 Main Street, #E, Venice, California 90291

A Robot Arms the pc



by David Hunter

As automation invades factories and manufacturing plants, and as we embrace the robot revolution, ease of use is a preeminent concern in the development of the next generation of industrial machines. Just as computers had to be made easier to use and understand, robot systems need to be more personal, easier to maintain, and operable by one person. IBM's 7535 Manufacturing System is one significant step toward realizing the dream of a low-cost, adaptable, easy-to-use, mass-produced robot.

The 7535 Manufacturing System is IBM's premier entry into the robot industry, although the robot arm itself comes from Japan (produced in Tokyo by Sankyo Seiki Manufacturing Company Ltd.). Programmable with the IBM Personal Computer, the 7535 is clearly focused on the growing assembly market.

IBM is no stranger to robotics technology; the company's marketing sense is uncanny, crystalline, born of massive resources, intelligence, and wisdom. IBM's entering the robotics field is as significant as, if not more so than, its entering the personal computer market. The name IBM will help to legitimize the robot industry.

Winning Friends. Three factors—reliability, ease of use, and adaptability—are the major concerns in modern assembly robot design. In robotics, reliability is measured by a robot's repeatability factor—how close the robot comes to the same spot of the work area in successive tries. The 7535 has a repeatability factor of plus or minus 0.002-inch and four degrees of freedom. A Z80-based unit with 6K of memory controls the 7535 and is capable of storing five different programs. Programming is done on an IBM Personal Computer using an IBM-authored language. In some applications, one Personal Computer can be used to program many 7535s.

The IBM 7535 is a general-purpose robot arm with various potential uses. Among these are automatic assembly, feed and insertion of odd-shaped parts, multiple-point drilling, tapping, chamfering, multiple-point soldering and screw tightening, assembling and inspecting parts in a process, packing products into cases in rows—or taking products out of cases, and loading and unloading parts on automated production lines.

Reliable estimates of growth in the use of industrial robots (*Robotics Age*, September/October 1981) project that robotics will be a two billion dollar industry by 1990. Assembly tasks are expected to account for 35 to 40 percent of robot orders in 1990, though they only account for a little more than 10 percent today. In 1990 General Motors expects to be using more than five thousand assembly robots.

"Assembly is a generic term. Very broad," explains Jim McDonald, division director of industrial automation at IBM's complex in Boca Raton, Florida. "It ranges from materials handling to inserting electronic components, testing, and fabrication. The 7535 can clearly do materials handling in addition to tasks like drilling."

According to *Robotics Age*, materials handling, including machine loading and unloading (categorized separately from assembly), will account for 30 to 35 percent of robot orders in 1990; materials handling accounts for 25 to 30 percent of today's robot market.

Quick Study. "It is possible to be productive with the 7535 after half an hour," McDonald explains. "The 7535 has no sensing and the language is not complex. It's a light assembly robot capable of handling smaller objects, like pieces for subsets in a large assembly line. The reception of the product has been very good."

McDonald, a nineteen-year IBM veteran, has worked in such diverse fields as word processing on the MTST system in the mid sixties ("There were very few word processing systems at the time") and the 6670 laser printer. McDonald also served a stint as administrative assistant to a group executive in New York City. He's been at Boca Raton for a couple of years now and finds the robotic division "exciting."

"Assembly is a new, young industry and IBM intends to compete aggressively," he says. "We'll also continue to examine the market."

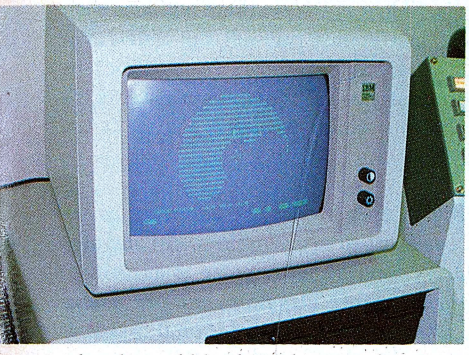
In terms of cost, the 7535 Manufacturing System is very com-

petitive. Starting at $28,500 (excluding costs of the Personal Computer and software), it's not something you'll want to give away as a Christmas present or buy to impress guests. On the other hand, Santa Claus sure could make good use of a few dozen.

Deaf, blind, bolted to a table top, IBM's 7535 is the essential production slave—tireless, accurate, unquestioning. Emerging from the country with the largest robot population, Japan, the SCARA (selective compliance assembly robot arm) type design is the work of Professor Mukima in a university north of Tokyo. Sankyo Seiki, manufacturer of the 7535, has marketed other SCARA-type robot arms in Japan. The 7535 was built to IBM's specifications.

Mister Muscle. The theta 1 axis arm swivel (the bigger yellow piece with the IBM logo on it—see photos), accomplished with a DC servo motor, has a range of 200 degrees and a variety of speeds, between twenty-three inches per second and fifty-seven inches per second depending on the weight of the object being moved. The theta 2 axis arm swivel (the smaller yellow piece of the arm), with its own servo motor, has a range of 160 degrees and moves at from fifteen inches per second to thirty-nine inches per second. The roll axis (the gripper in the photos) has a range of 180 degrees and a weight limit of 13.2 pounds.

Looking at the specifications is one thing, but seeing the 7535 in action is even more dramatic. It makes one respectful of these deaf, dumb, and blind machines. The 7535 is fast. It's not pretty or glamorous. It gets the job done, pronto.



Left: Monitor display when the 7535 Manufacturing System is in the teach mode. The cursor in the middle of the screen indicates the position of the arm in the work area. Right: Jim McDonald, division director of industrial automation at Boca Raton. Facing page: Will Summers of IBM's system products division puts the grip on the 7535's end-of-arm effector.

Housed in your typical rectangular metal box about waist high, the 7535's Z80-based controller sits to one side, connected to its operator panel and the Personal Computer. A point-to-point system is used for positioning control; the controller's specifications include fourteen external input signals and fifteen external output signals.

The IBM Personal Computer with an IBM 80 CPS Matrix Printer, IBM's DOS, and AML/E (an entry level version of IBM's in-house language AML) acts as the separate programming unit. No special hardware additions are needed, other than an asynchronous communications adapter for transmitting information to the 7535.

Down and Out. Once the Personal Computer downloads the program to the controller, its job is done. A shrewd idea. The versatile Personal Computer can then be used for further programming development or for some completely different task such as word processing or accounting.

Multiple 7535 systems programmed by a single Personal Computer would be ideal for batch processing (usually a labor-intensive production where a small number of products are made in a short period of time). Each robot would serve a work cell. In a batch process for manufacturing gears, for example, the first robot would take a gear, run it through a dual spindle vertical chucker, and then pass it to the second robot. The second robot would put the gear through a gear shaper and deburr it before passing it to the third robot. This last 7535 would put the gear through a shaver, and so on.

The advantages inherent in IBM's situation are considerable. They have enormous resources, money and research facilities, not to mention manufacturing plants for in-house applications and testing. IBM also has the largest line of computers and related accessories in the world. They can provide the brawn and the brains needed for robotics in a variety of forms.

IBM's own boxframe robot, the RS 1, is being aggressively test marketed right now (see accompanying sidebar). A more complex and ambitious assembly robot than the 7535, the RS 1 has six degrees of freedom and some senses. It is a product of evolution rising from an ocean of in-house robotic development at IBM that dates back to 1972.

"The idea is to make everything compatible with everything," McDonald begins. "The RS 1 is programmed with the IBM Series/1 computer. The 7535, which doesn't require as much computing power, is compatible with the Personal Computer, and, ultimately, with any IBM personal computer."

Short and Sweet. IBM's robot language AML (A Manufacturing Language) was developed in-house specifically for industrial automation. The 7535 version, AML/E, is considerably simplified because of the lack of visual or tactile sensors in the robot arm and in the end-of-arm effector. AML/E features commands to position the jointed arm at x-y coordinates in the workspace, to manipulate the end-of-arm effector, and to communicate with other devices through input/output ports.

Programs are short, comprised mainly of subroutines. A program may feature statements and subroutines, all ending with semicolons. Statements may be variable assignments or commands. The following sample program is for a routine assembly task: picking up an object, moving it to a new point, and putting it down. Comments describing the function of each program line are preceded by dashes (—).

```
NEWPROG: SUBR; —beginning of program
FEEDER: NEW PT(−300, 400, 60); −point declaration
BELT: NEW PT(−650, 0, 0);
PICK: SUBR(A): —internal subroutine passing the parameter A
PMOVE(A); —go to point A
DOWN; DELAY(.5); GRASP; UP; −pick up an object
PMOVE(650, 0, 0); −go to home position
DOWN; RELEASE; DELAY(.5); UP; —set down the object
END; —end of subroutine 'PICK'
LABEL:ITERATE('PICK',<FEEDER,PT(0,400, 0), BELT>);
  −call 'PICK' three times, passing each of these three points
WAITI(8,1,6); —wait six seconds for input port #8 to become 1
WRITE0(14,0); −send a value of zero to output port #14
TESTI(1,0,SKIP); −if input port #1 is zero then go to skip
BRANCH(LABEL); —go to the iterate step
SKIP: END; —end of program
```

This sample program, provided by IBM, demonstrates all the standard manipulator, sensor, and logic commands. The words in AML/E are not far removed from the ordinary English words describing the robot's actions (move, grasp, release). The language seems easy enough to use once you have an idea of what you want the robot to do. Still, you can't help but dream of the day when you can say to a robot the equivalent of "Get the little sucker and flip it over there."

Quiet Revolution. AML/E and its parent, AML, seem like

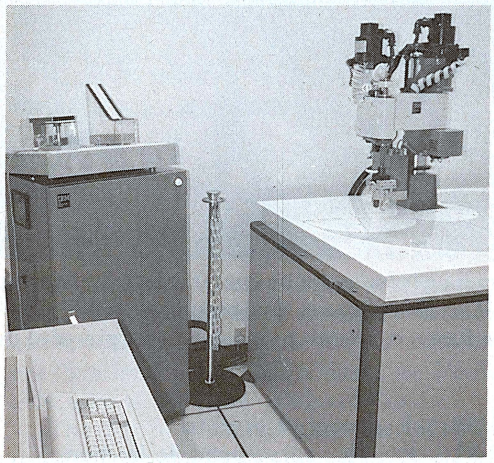
modest accomplishments compared to IBM's achievements in other fields, but, like the 7535 Manufacturing System itself, they signify a development in the robotics field long awaited—a standardized approach to the industry.

Another extremely helpful feature of the 7535 system is the teach mode. In this configuration, the Personal Computer talks directly through the controller to the robot; the robot in turn responds to keyboard input.

When the 7535 is in the teach mode, the monitor displays an overhead view of the work area (see photo). The cursor marks the x-y position of the end of the arm. Using the four cursor keys on the numeric keypad portion of the Personal Computer's keyboard, you move the cursor on the screen and the robot obligingly follows along. In this mode, it is possible to move the arm very small distances (0.005-inch) for precise positioning. This makes it possible to walk (or crawl as the case may be) through the steps of an application.

The area immediately surrounding the intersection of the x and y axes is darkened in the screen graphics. If you move the cursor into that area, the arm stops; because of the construction of its own body, it can't move further. When you move the cursor back into the live work area, the arm moves to that spot within a second or so.

IBM has not shipped any 7535s yet; that will begin to happen in the fourth quarter of this year. Training courses for customer-maintenance personnel start in the third quarter. Three-day courses to be held at Boca Raton will begin with an introduction and overview of the 7535 system, followed by systems operations, adjustments, replacements, and hands-on trouble-shooting.



The complete 7535 Manufacturing System. The Personal Computer is at the lower left corner with printer and monitor (not in the picture). The Z80-based controller is at the left with its operator console (console not pictured). The robot arm is bolted to a work table.

The 7535 stirred interest at its public unveiling at Robots VI, an industry convention sponsored by Robotics International of the Society of Manufacturing Engineers and the Robot Institute of America. Held during March in Detroit, Robots VI was a landmark event (not the least exciting thing being IBM's announcement), drawing more than twenty-seven thousand attendees.

Three Cheers. IBM's entry into robotics is good news for the world's manufacturers. The robot industry may reach a growth rate of 35 percent a year through this decade, according to *Robotics Age*.

General Motors and Westinghouse are two other giants entering the market along with IBM, and this trend is not such a good one for the small robot companies in existence today. In the next few years a shakeout will occur, with the less competitive firms dropping out.

Venture (June 1982) quotes a supplier as saying: "If I were a robot manufacturer I would be running scared. There's going to be a bloodbath." Robot wars already?

Any way you look at it, IBM could practically swallow up the entire robot industry, which is expected to reach sales of $200 million this year—a paltry sum to a company that grossed $29 billion in total sales last year. Before any one company can take it over (an unlikely event), the robot market has to grow a little, and obstacles are present.

McDonald firmly believes that a lack of trained people is the major ailment holding back growth in the robotics industry.

"Robotics combines a lot of different skills: industrial engineering, mechanical engineering, and application programming. The robot applications engineer must have a broad skill base.

"The best way is to have the customer himself acquire the expertise. We need to have a broad base of development people with experience and education. Once we solve that problem, the enormous opportunities of robotics will open up."

House of Mystery. There is little question that IBM will have more robotic products in the future. What they might be and when we might see them remains a mystery for now.

"From an assembly standpoint, it's desirable to have in a robot as many different applications with different sensor capabilities as possible. We've leaned heavily in that direction," McDonald offers.

Centered for the moment in steamy Boca Raton, about an hour north of Miami, IBM's robotics department occupies only a fraction of the enormous facility. Boca Raton (which means "mouth of the rat" in Spanish) is a small town. In summer, the humidity and considerable heat combine to make life nearly unbearable anywhere outside IBM's beautifully designed air-conditioned buildings.

A burning question that invariably gets asked when people are discussing the robot industry is, "How many workers will lose their jobs because of robots?" One fact stands out—all the blue-collar workers needed in the year 2000 are already born. In true Socratic fashion, sources in the robot industry ask: "How many parents today are raising their kids to be blue-collar workers? Where is the work force going to come from?"

Robots, you betcha!

Unions are already moving to ensure that the adoption of industrial automation does not cause massive unemployment and economic stress. Ford and the UAW have recently come to agreement on a joint Union-Management Employee Development and Training Program. Other unions, like the IAM (International Association of Machinist and Aerospace Workers) and the IUE (International Union of Electrical, Radio, and Machine Workers), are investigating the nuts and bolts of contractual agreements for workers who are displaced by technological change.

Kicked Upstairs. Ideally, if a worker loses a job to a robot, then he or she is upgraded to a higher position. By the year 2000 there won't be enough humans to man the factories, but those who still remain will be reeducated for other positions.

Diehard science fiction fans might remember the June 1954 cover of *If* magazine. It sports a hundred-foot-tall humanoid robot, illustrating Walter M. Miller's story, "Check and Checkmate." The metallic giant has a sackful of humans slung over one shoulder. With one huge claw, the robot is sowing humans in a field, like a farmer. To dispell all doubts about what is going on, the artist, Anton Kurka, placed a human-looking barn in the background.

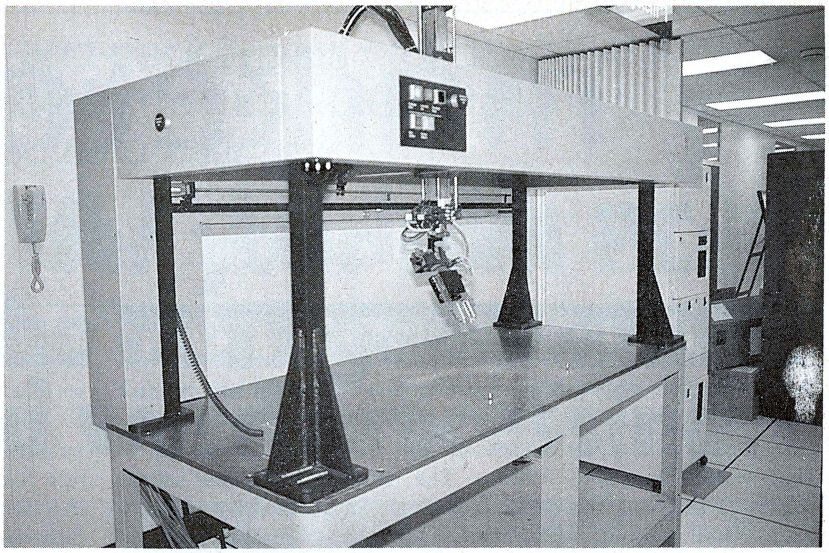
A bit far-fetched you say? Well, of course it is. Still this ludicrous image of robot omnipotence and inhumanity says a lot about how many people still view the subject. Education is needed to deflate the unrealistic mythology surrounding robots. Education is also needed if individuals are going to survive in a work force where knowledge of and experience with advanced technology are becoming increasingly crucial.

The world of industrial robotics is not yet very exciting to your average *Star Wars* aficionado, but it's only recently come into being. Ten years from now, robots will be in manufacturing facilities all over the world, and they'll probably find their way into ever increasing numbers of homes as well.

Will there ever be a personal robot in the way that there are now personal computers? McDonald smiles.

"Two years ago someone asked me that and I said, 'I don't think so.' Now I'm not so sure." ▲

A Brainier Robot



IBM's RS 1 Manufacturing System is a product of evolution rising from an ocean of in-house robotic development at IBM. The manipulator (center of picture) has senses and six degrees of freedom, and is very fast, traveling anywhere in the work area within one and a half seconds.

Robots have come a long way. Thirty years ago they were just a dream—the bastard children of visionary science fiction writers. Nowadays, the dream's promise has been partially fulfilled by complex industrial robots capable of performing a myriad of tasks faster, better, and longer than a human could. Robots aren't quite unquestioning, untiring personal slaves, capable of mixing a mean martini and providing companionship. Not yet. But progress made in the last two decades does point to an exciting future.

IBM's experience with robotics dates back to the early seventies. It all began at the Yorktown, New York, research facility; throughout the seventies, in-house robot systems began to appear at IBM manufacturing sites around the world.

The Boca Raton group was formed in 1978 with the goal of eventually putting IBM into the robot market. The first IBM-produced robot to see the light of day was the RS 1 Manufacturing System, now in the advanced stages of test marketing at IBM manufacturing sites and selected outside locations.

Missing Link. "This system ties it together," explains Jim McDonald, division director of industrial automation at Boca Raton. "The RS 1 is the result of an evolutionary process. It's an outgrowth of our goal of creating an adaptive robot."

Considerably more complex than IBM's deaf, dumb, and blind 7535 (manufactured by a Japanese company), the RS 1 features sensors and six degrees of freedom. Strain gauges located in the gripper provide force-sensing capabilities at the side, tip, and pinch surfaces of each finger. The presence of objects between the fingers can be detected with a photoelectric cell.

The RS 1 is housed in a rectangular metal frame. The work space is eighteen inches deep by fifty-eight inches wide by seventeen inches high. The manipulator can go anywhere in the work area at a top speed of forty inches per second. The manipulator has 270 degrees of roll, 180 degrees of pitch, and 270 degrees of yaw. The manipulator moves at a top speed of 180 degrees per second.

AML (A Manufacturing Language), developed by IBM, is used to program the RS 1; the language features commands such as *approach*, *withdraw*, *transport*, and *grasp*. A modified IBM Series/1 computer system is the controller and features valuable data processing functions like calculation, printing, and disk and diskette data management. A hand-held control box called a teach pendant may be used for manually moving the arm and defining joint positions during application development.

Processing Pals. Designed for precision assembly, the RS 1 is being used at IBM plants in putting together word processor units and inserting gears and ribbons into type cartridges. According to IBM, the RS 1 and 7535 are capable of functioning together. The 7535 would unload and pass parts to the RS 1, or take finished products from the RS 1. Soon the two machines may be able to build each other. Well, not that soon.

Fast and strong, the RS 1 is very sensitive and capable of delicate maneuvers.

The famous egg incident is a good way to show how sophisticated a machine the RS 1 truly is. One day a gentleman was shown a demonstration of the RS 1. After having lunch, the gentleman came back with a dozen eggs. He said: "I want to see the robot pick up an egg and move it." There's nothing like a good challenge.

On its first try, the RS 1 picked up the egg, but when it moved the egg fell. The second time, the RS 1 successfully picked up the egg, moved it, and set it down. IBM technicians saved that program on disk for future doubting Thomases. ▲

Get MORE out of VISICALC With

V-UTILITY $99.95

COPYRIGHT © 1981 YUCAIPA SOFTWARE Version 5.0

NOW AVAILABLE FOR IBM PERSONAL COMPUTER (PC DOS)
APPLE II (DOS 3.2 and DOS 3.3)
TRS-80 MOD I, II, III and 16 (TRSDOS)

V-UTILITY CONSISTS OF ALL THESE VISICALC "USER FRIENDLY" UTILITY PROGRAMS ON ONE DISK RUN BY INDEX AND PROMPTS.

V-PRINT WITH THIS PROGRAM YOU MAY SELECT THOSE COLUMNS YOU WANT TO PRINT ON THE PRINTER AND PLACE THEM IN ANY ORDER YOU SELECT. A FLEXIBLE PRINTING UTILITY THAT ALSO WILL PRINT SHEET EQUATIONS

V-STAT PROGRAM COLLECTS DATA AUTOMATICALLY FROM THE VISICALC COLUMN AND CALCULATES NUMERICAL DISTRIBUTION, CORRELATION COEFFICIENT, REGRESSION ANALYSIS, CHI² TEST, AND T-TEST. YOU MAY SELECT THE COLUMNS FOR DATA ENTRY AND SPECIFY THE ROW# TO START AND ROW TO END DATA COLLECTION. PROGRAM IS EASY TO RUN.

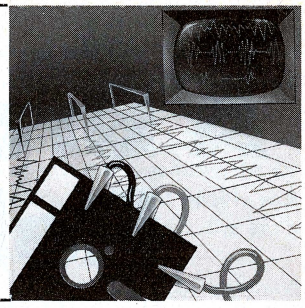
V-PLOT THIS PROGRAM AUTOMATICALLY INPUTS DATA FROM A VISICALC COLUMN, PERFORMS AUTO SCALING THEN PLOTS EITHER 1 OR 2 COLUMNS ON A REGULAR LINE PRINTER (GRAPHICS NOT REQUIRED). IDEAL FOR ANALYZING UP TO 250 NUMERICAL DATA POINTS IN RELATION TO TIME.

V-OVERLAY PROVIDES THE FOLLOWING SELECTION OF OVERLAYS TO LOAD ON TO THE VISICALC SHEET. MOVING AVERAGES, EXPOTENTIAL SMOOTHING EQUATIONS, TIME SERIES TREND ANALYSIS, DATE COLUMNS.

YUCAIPA SOFTWARE
12343 12TH ST · YUCAIPA·CA·92399
PHONE (714) 797-6331 ALL PROGRAMS AVAILABLE SEPARATELY $39.95 EACH

IBM, APPLE, TRS-80, and VISICALC are trademarks respectively of International Business Machines Corp, Apple Computer Inc, Tandy Corp, and VisiCorp

marketalk reviews



Unless otherwise indicated, software reviewed runs in DOS on machines with either display adapter and requires 64K and at least one disk drive.

Volkswriter. By Camilo Wilson. The name suggests one salient quality of this word processor: it's meant to be used by ordinary folk. Wilson has gone out of his way to avoid technical language in the documentation and prompts. He's also been lavish with on-screen help. The program disk includes three tutorial files that do such a good job of teaching the program that you truly do not need the documentation. While you're working, you can hit F1 and get a display of the program's twenty function-key commands. Unintentional erasure of files or loss of data in RAM is more than adequately discouraged through the use of confirmation prompts.

Editing is screen oriented in *Volkswriter*, meaning that you're free to wander all over the screen as you write and change whatever you please; there's no separation of editing and writing functions. The program retains the standard DOS line editing procedures and adds vertical mobility with the up and down arrows and the page keys.

As far as output is concerned, what you see is almost what you get. At the time you create a document, in the edit mode, you define tab stops and the position of the right margin. So the way the lines break on the screen is the way they'll break on paper. Should you change your mind during the course of an edit, you can redefine the right margin and hit the F8 key to rewrap each paragraph.

Certain other output parameters are specified at print time. There you'll encounter a format menu in which you choose between justification and ragged right, order up page numbers if you desire, and select line spacing. You also control the left margin from this menu. Frequently used print formats can be saved in their own files and called up by means of an embedded command at the top of your document. The same embedded command allows you to change formats within a document, so you can, for example, have single-spaced and double-spaced text within the same document.

There are only two other embedded commands; one facilitates the printing of large documents by allowing you to chain files together, and the other generates a pause in printing, so that you can chain files that reside on separate disks.

You're likely to find the file-chaining commands handy, because *Volkswriter* uses a lot of memory. On a 64K system, with the editor loaded, you'll have about 13K left over for writing. Lifetree recommends you have 128K, but you can get by with 64. You'll just need to keep your eye on the figure in the lower right corner of the screen that tells you what percentage of the available memory remains. And beware: the *Volkswriter* printer module uses more memory than the editor, so a document that shows 25 percent memory available in the editor may turn out to be too big to be printed.

Fortunately, the program is well supplied with block-move functions. You can mark an area of text with two strokes of the function keys and then delete or move the block—or save it separately on disk. So if you accidentally exceed the memory limitations, it's relatively easy to recover. The block-move functions also make it simple for you to move a paragraph from one section of a document to another, or to zap a chunk of text with a total of three keystrokes.

Volkswriter will not meet everyone's word processing needs. It will not, for example, supply your document with running heads or feet. It's not a convenient tool for jobs that require frequent changes in indentation. It does use the IBM/Epson printer's boldface capability—but only on whole lines. It does not provide for underscoring.

The program's extensive fail-safe procedures may prove tiresome to some users. What's annoying is not just that you have to confirm every move that could conceivably leave a file stranded in RAM; it's that every time you make such a move, *Volkswriter* runs out to disk to retrieve a little warning message. Furthermore, the keyboard buffer is disabled, so you need to wait patiently for the warning before you can give the go-ahead sign. It would be nice if the program had provided an expert-user mode to make these delays optional. CS

Volkswriter, by Camilo Wilson, Lifetree Software (177 Webster, Monterey, CA 93940; 408-659-3221). 128K recommended. $195.

VEdit, by Theodore Green, and **FSEditor**. Program tools designed to create and change text on a computer come in two varieties: word processors and program editors.

Writing programs is different from writing letters and reports, and program editors differ from word processors in the way they look at text and in the commands and tools they give the user. Word processors are focused mostly on words and paragraphs and have features such as automatic wraparound from line to line; program editors are more likely to treat lines as distinct units and have programming-oriented features.

There's a lot of middle ground between word processors and program editors, and a flexible program of one kind can serve well for both purposes. People doing a lot of programming, however, may well want to buy a program editor, even if they already have a word processor.

Both *VEdit* and *FSEditor* are reasonably easy to use but, like most other complex tools, take some learning and getting used to. Both are full-screen editors, and both make good use of the pc's ten function keys. They both come with good documentation, and both have a few bugs. That's about where the similarity ends.

FSEditor has lots of menus, help displays, and general on-screen razzmatazz. This is helpful to the beginner but becomes a nuisance for an experienced user. *FSEditor* takes a conservative approach to safeguarding your work in progress; it keeps a temporary working copy of the edit text on disk, so if the computer stops for any reason, you won't have lost much work. This fail-safe provision slows the program's operation but does indeed protect your effort.

FSEditor is inflexibly line-oriented; practically every command treats lines as units, and there's no word-wrap feature at all. The function keys are defined by the program and can't be changed; like Basic, *FSEditor* gives a summary of function keys at the bottom of your screen.

FSEditor makes heavy use of the speaker, as well as the bright and blinking features of the screen display. While this should get good marks for intelligent use of the computer's capabilities, much

Now even hunt-and-peck typists produce perfectly formatted letters...memos...presentations... scripts...*automatically.*

POWERTEXT® for your IBM PC: Announcing the next generation in micro-based word processing.

"A business professional wants to think about what he or she is writing. Not worry about getting the words in the right places on the printed page.

"That's why we developed a new kind of high-performance system that handles printed style automatically.

"PowerText knows exactly how you want your letters to look...your memos...reports...every kind of document you write.

"You define the formats. PowerText stores them in its style files.

"When you write, just bat out your copy on your IBM PC. Forget about printed style.

"Do all the editing you want.

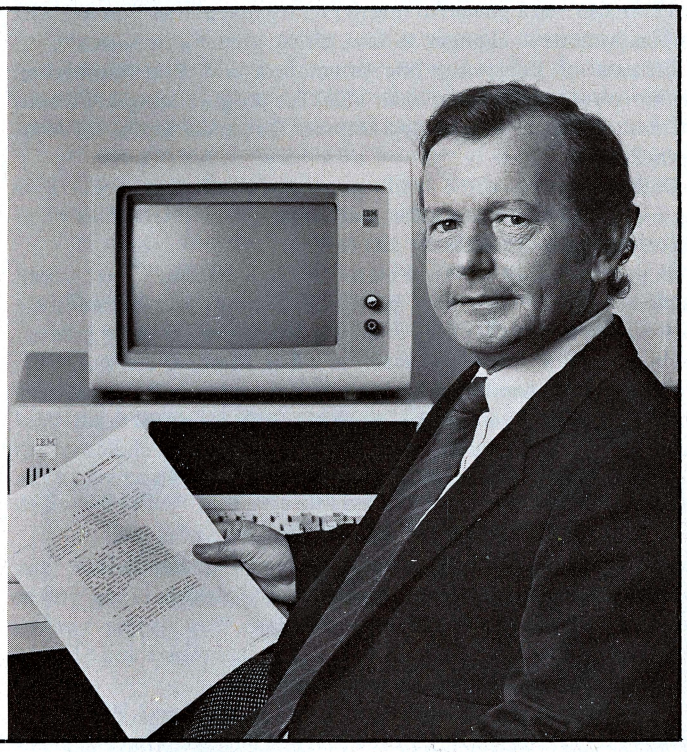
"Then...just tell PowerText 'Make this a letter,' and you get a letter...'Make this a sales presentation,' and you get a sales presentation...

"It's almost like dictating to your computer. Even if you're an unskilled typist, you get a beautiful, perfectly formatted document...every time.

"And it's all very straightforward...very easy to use.

"We think PowerText is the way a word processor is supposed to work."

David P. Guest, President, Beaman Porter, Inc.



All this is automatic in every kind of document you write:

* Margins
* Indents
* Spacing
* Title page
* Envelope, label
* Intelligent page breaks
* Justification
* Centering
* Variable pitch
* Pagination
* Table of contents

Plus many other high-performance features:

* Complete editing capabilities
* Form letters
* Automatic headers and footers
* Boxed copy
* Print macros
* Up to 14 columns
* Automatic footnote numbers
* Boilerplate inclusion
* Vert. and horiz. border lines
* No limit to document length
* 132-character lines
* Superscripts and subscripts
* ...and much more

25% more data per disk

PowerText accommodates 200K bytes per disk, 25% more data than the 160K bytes accommodated by standard IBM software.

System requirements

PowerText is a run-time, bootable system. It requires at least 64K of memory, two disk drives, and a printer. It supports both serial and parallel printer interfaces, extended memory, and RAM disk.

Full 5-year warranty

The system is warranted for a full 5 years. Should it ever fail to perform as specified, Beaman Porter, Inc. will fix it at no charge.

Available from selected dealers
Or order directly from Beaman Porter.

VISA and MasterCard accepted. NY State residents add appropriate sales tax.

Complete software $399
Includes diskette plus excellent manual/tutorial
Manual/tutorial alone is $25

PowerText®

BEAMAN PORTER, INC.
High Performance Computer Products

Pleasant Ridge Road, Harrison, NY 10528 (914) 967-3504

of it can quickly get tiresome. On the other hand, *FSEditor*'s use of special graphic characters is imaginative and helpful.

FSEditor is copy protected.

VEdit, in contrast, is bald and plain in operation. No menus, no help screens. This may be a disadvantage for the occasional user, but it's a real boon to anyone who uses *VEdit* regularly; there's nothing to get in the way between you and your text. Although *VEdit* is largely a program editor, it does have a number of word processing features such as word-wrap and paragraph reformatting. *VEdit* keeps its working copy of the text in RAM, where it's vulnerable to power failures; but, as a consequence, it's faster than *FSEditor* and almost never keeps you waiting.

VEdit is not copy protected. In fact, it comes with a customization program that lets you decide such things as tab settings, indentation, and the use of the function keys. You can produce several different versions of the program if you wish, each tailored to a particular use.

VEdit also has a powerful macro facility that allows you to make complex automatic changes to text, either globally or selectively.

Because of its strong line orientation and line numbering, *FSEditor* would work better than *VEdit* for programming in Fortran or Cobol. *VEdit* has indentation features that make it nice for writing in Pascal.

Both programs come with good documentation. *VEdit*'s is lengthier, with more examples and explanation. But *FSEditor*'s is clearer and easier to look up answers in.

In general, *VEdit* seems to be the stronger product. First, it can double as a poor man's word processor. Second, its commands are more powerful and flexible. Finally, it's a stable, mature product (it's been around for a few years on other personal computers).

On the other hand, if price is crucial, if menus and help screens are important, or if you're looking particularly for a tool to edit programs in Fortran or Cobol, then *FSEditor* is a better choice. pn

FSEditor, Cypress Software (Box 8715, Coral Springs, FL 33065). $125.
VEdit, by Theodore Green, CompuView Products (1955 Pauline Boulevard, Ann Arbor, MI 48103; 313-996-1299). $195.

Everest Explorer. By William Godwin and Don Knowlton. You have a vision: the greatest mountain in the world is luring you to challenge its bleak slopes and awesome peaks. You also have one or two hundred thousand dollars cash to recruit climbers and Sherpas and to buy the equipment you will need.

Is two to one a good ratio of Sherpas to climbers? Would 250 bottles of oxygen be adequate for a party of 30 men? There is no one to help you with these questions, and the fate of the expedition depends on your answers.

Next you fix your departure date. A little bad luck here and you waste two weeks sitting in camp waiting for the winter storms to abate, or, worse, miss two weeks of prime climbing weather and get caught later on by the summer monsoon.

There are two routes up from base camp—the difficult southern ridge and the even more difficult western ridge. (You are not even allowed to try the risky north ridge or the vertical south face, although both have been conquered by less timid mountaineers.) First it is necessary to clear the avalanche-swept Ice Fall route and establish Camp 2. That done, the real climbing begins: climbing *down* to Camp 1 to pick up supplies, then back up and so on, until you feel ready to establish Camp 3. There are six camps in all. Most of your time is spent running back and forth with supplies. The chain of camps is long, and a lot of supplies get consumed en route. It takes three Sherpas hauling from Camp 1 to Camp 2 to support two hauling from Camp 2 to Camp 3, and so on. Camp 6 may be nothing more than a couple of climbers huddled in a single tent. The logistics are very challenging for the average pc owner, though he might benefit from reading *Everest the Hard Way* by Chris Bonington. (This was an important source for the game designers. It is only fair to let the players look at it too.)

You need to determine who is carrying what to where everyday. Oops! Better not give such a heavy load to Pasang Tenzing Sherpa. He's feeling tired today and also has frostbite. To complicate things even more, your heroic climbers keep getting altitude sickness or hypothermia. They get buried in avalanches or simply disappear without a clue. Sudden spring storms send them to their tents coughing.

Watch out! Too many injuries and the Sherpas may decide to abandon the expedition, which means that you might as well abandon it too.

To play a game like this is almost like climbing the mountain itself. Look forward to eight or nine hours of telling Pasang Tenzing Sherpa to run up to Camp 2 and back again. The keyboard buffer is disabled, which means you must continually wait for prompts while entering all these repetitious instructions. Alas, there is no breathtaking scenery to alleviate the tedium; the game has absolutely no graphics.

Given all of this, why would anyone want to play *Everest Explorer*? Maybe "because it's there." fj

Everest Explorer, by William Godwin and Don Knowlton, Acorn Software Products (634 North Carolina Avenue S.E., Washington, D.C. 20003; 202-544-4259). $19.95.

Real Estate Investor. By Phil Rozansky. *Real Estate Investor* provides you with *VisiCalc* templates to use as a tool to evaluate real estate. Whether it's a single family residence or a multiple dwelling unit you're interested in, they've got you covered. As advertised, it's not a stand-alone program, but one designed to be used in conjunction with *VisiCalc*.

The templates come complete with sample assumptions. While you are reading through the documentation to familiarize yourself with the package, you can calculate the models presented and print out the results. When you are ready to evaluate a possible purchase of your own, simply enter the new information in the locations indicated, recalculate your new model, and print out the results. This package is especially helpful if you are considering several different financing packages. You can enter the finance terms of each package and compare the printed reports to aid you in reaching an intelligent decision.

Two supporting schedules are also provided for a more detailed analysis of income and expenses. The program was designed around the tax guidelines of the Economic Recovery Act of 1981.

The manual carefully explains all the sections of both the individual residence and income property models. It also includes a chapter that describes the proper entry of various financing alternatives. New wrinkles in financing appear from time to time, so all of the latest schemes may not be represented, but the more commonly used financing methods—like conventional mortgages, assumed mortgages, full-balloon mortgages, partial-balloon mortgages, and variable-rate mortgages—are handled quite nicely. A chapter reviewing the material you need to know from *VisiCalc* is also included. The appendix contains sample reports for your reference.

If you're already an avid *VisiCalc* user, you may doubt the necessity of purchasing this program instead of developing the necessary templates yourself. Once you see the scope and complexity of many of the equations involved to produce the reports, however, you will agree that it's nice to have all that hard work already done for you so you can get on with the business at hand—making money! sf

Real Estate Investor, by Phil Rozansky, Simple Soft (480 Eagle Drive, Elk Grove, IL 60007; 312-364-0752). $129.95.

Millionaire. By Jim Zuber. You have $10,000. You hope to parlay your savings to a million dollars. The method you have chosen is not exactly original—you intend to speculate on the stock market. No long-term investment, no counting dividends—this is pure speculation.

The market is simplified. There are only five kinds of indus-

tries—computer (naturally), oil, retail, auto, and heavy—and three stocks in each category. You can invest in IBM stock but not in Apple, in oil but not in solar energy.

Market changes are reported weekly, and the game lasts for ninety-one weeks. You can change your portfolio each week. There is a graph for prices on the market, and also for each industry and individual stock, to help you make decisions. In addition, you can read *The Micro-Z Journal* to pick up tips and information. Price fluctuations seem realistic, though it is sometimes surprising to see a staid blue chip fall to the penny-stock level or triple its value in a short period.

As you increase your portfolio, you gradually become a more expert speculator. You buy on margin or deal in stock options. Having amassed $100,000, you can borrow against your assets, risking money you don't own. The program keeps track, in a rudimentary fashion, of taxes, broker fees, and interest on borrowed money. But other important factors are ignored. There are no dividends. Money in the bank brings no interest. There are no stock splits. There is no distinction between common and preferred stock. There is no inflation.

The programmer spent a lot of effort making the graphics neat, and if you have a color adapter, you will be pleased with the effect (the game will also run in monochrome). However, the attention to graphs and charts would have been better spent on making the game a more realistic market simulation. As it is, the program focuses too narrowly on one aspect of investment.

Nevertheless, *Millionaire* does drive home some of the old saws: "Buy low and sell high," "Nothing was ever done without patience," and "A fool and his money are soon parted." It is a fine diversion for the vicarious tycoon. fj

Millionaire, by Jim Zuber, Micro-Z Applications (22704 Ventura Boulevard, Woodland Hills, CA 91364; 800-835-2246, ext. 101). $49.95.

Championship Blackjack. By Joe Juhasz and Keith Oswald. Blackjack is the only casino game in which the player can enjoy a theoretical advantage over the house. That's because the odds change as the deck is played out. At times the odds favor the house, and at other times they favor you. If you know when probability is smiling, and when she frowns, you can vary your bet—and your play—in such a way as to give yourself a small edge.

This program will teach you how.

Championship Blackjack is not only for card counters, though. It will also show you how to play correct basic strategy, without counting cards. If you do no more than learn a dozen or so easy rules, you'll be able to entertain yourself for hours at a blackjack table without the likelihood of any greater disaster than you might experience at the nickel slots.

The program runs on either display adapter. In mono you get six seats at the table and in color only five; in other respects the two display modes are identical. The cards are simulated neatly and almost realistically through text characters.

There are a lot of ways to play *Championship Blackjack*. A rules menu allows you to control such factors as the number of decks used, the number of cards actually dealt before the program reshuffles, the availability of surrender, whether the dealer stands on or hits soft seventeen, whether you're permitted to double on split pairs, and so on. Virtually every rule variation played in this country is available, including the option to double down after three or more cards (currently offered at one downtown Las Vegas club) and the new double exposure game, where both the dealer's cards are dealt face up but you lose on all ties. This range of options is a real value for both the serious student of the game and the casual player who wants variety.

You have other choices besides the variations in rules. You can play as many seats as you like yourself or you can fill the table with computer-generated players; you may also mix human and machine players. The computer will play either basic strategy or a card-counting system, according to your specification.

At any point during play you can hit F7 and get a display of the status of the deck—the running count of high cards versus low cards played, and the running count adjusted for the number of cards remaining to be played. Alt-F7 will cause the program to display the count continuously.

There's a simulation mode that allows you to check out the efficacy of the two strategy options. You can ask the program to run as many as 99,999 hands (about two weeks' worth if you have six players at the table) and then look in on the game every so often to see how it's going. Hitting F5 will get you a statistical report, including such information as the players' high and low bankroll points, their longest winning and losing streaks, the total amount wagered by each, and the percentage of the total wager won or lost.

There's even a tournament mode that enables you to compete against other players (human or machine) as well as against the house, according to the conventions of Las Vegas blackjack tournaments.

From the standpoint of the serious player, *Championship Blackjack* has only two flaws. The more serious is that during play the program gives you no information about how many cards in a deck have already been played. In a real game you'd be able to watch the discard pile grow and estimate the number of cards remaining to be played. Without that visual cue in this program you have no alternative but to keep a separate count of cards played.

The minor drawback is that the program's counting strategy, that of Dr. Julian Braun, is not by reputation the most powerful one ever devised. To correct that deficiency, however, PCsoftware is making available a separate utility (for $15) that will allow you to type in alternative strategies. CS

Championship Blackjack, by Joe Juhasz and Keith Oswald, PCsoftware (4155 Cleveland Avenue, San Diego, CA 92103; 714-279-2482). $34.95.

▲

FMS II Now Available for PC Users!

Get FMS II (Financial Management System II), the fast and easy-to-use small business and home accounting system, with Checkwriter for only $89.95!

Be able to enter an entire month's checking, charge card and cash accounts in just a few minutes using your own personalized macro list. INSTANT ERROR CORRECTION. Audit all files by code and month with year-to-date totals. PERFECT FOR TAX ACCOUNTING.

Self Prompting Error Avoiding Entry System, 1 to 3 key stroke entries with automatic date, coding and numbering.

Screen and Printer Routines for listing files, balance, reconcile, search and audit reports. Configure system to match almost any printer.

Search all files with specific and expanded searches.

Budget Manager to plan, review and balance your budget and generate complete reports for any 1 to 12 month period.

Account Auditor totals all files by code for any 1 to 12 month period with year-to-date totals.

Checkwriter prints your checks directly from FMS II files.

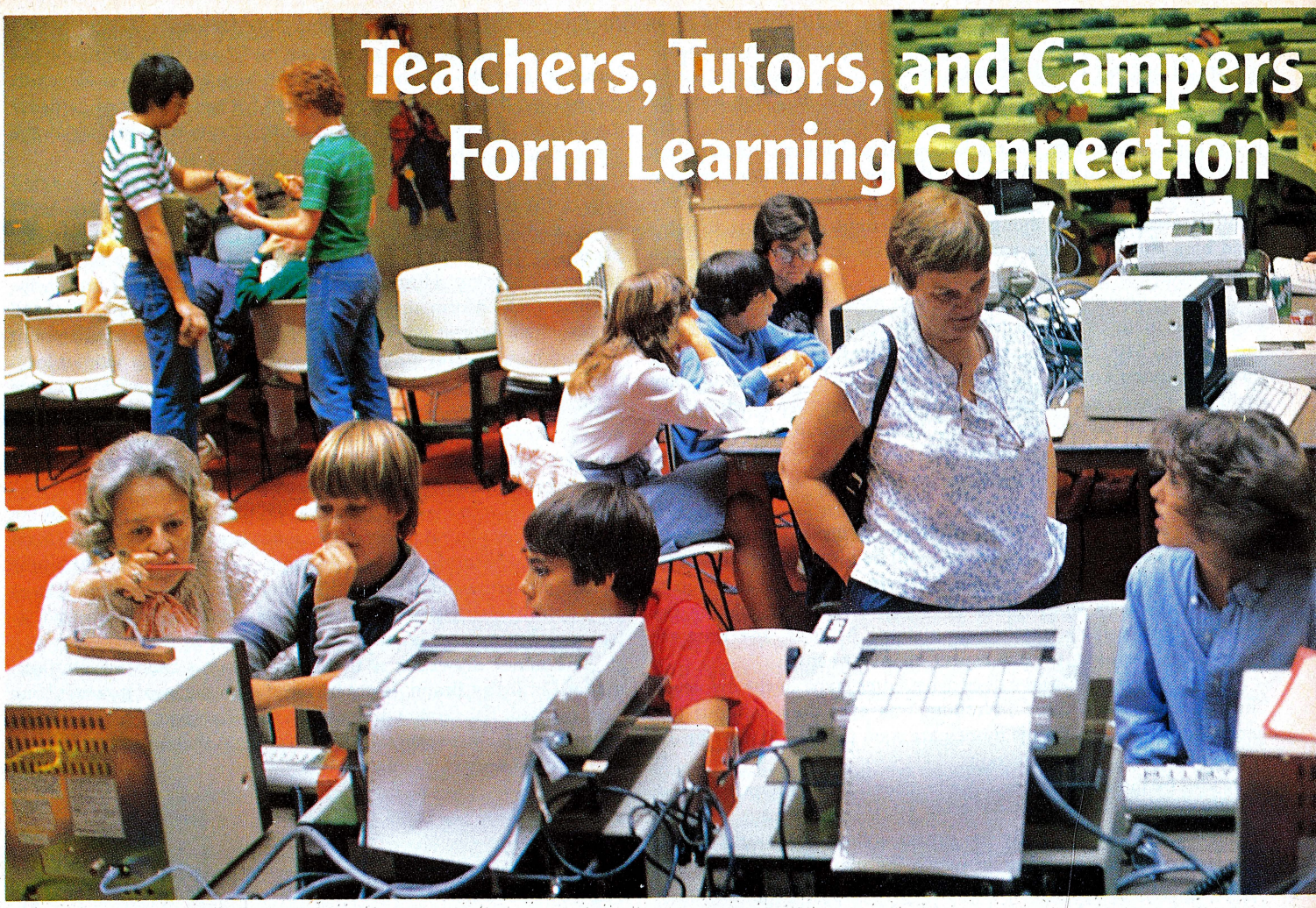
Item and Code Macros can be personalized up to 100 each.

FMS II requires 64K of RAM and Disk. Single or Dual Disk compatible. (Printer Optional) Price **$89.95.**

COMPUTERIZED MANAGEMENT SYSTEMS
1039 Cadiz Drive • Simi, California 93065 • (805) 526-0151
Check, VISA or MasterCard accepted. Calif. residents add 6% sales tax.
Dealer Inquiries Invited.

fmsII

Teachers, Tutors, and Campers Form Learning Connection



Tutors teach teachers and teachers teach campers how to use computers in Microcomputers in Education.

by Jean Varven

This summer Angela Randle of San Jose, California, Matt Baker of nearby Los Gatos, and Carlene Waldrum of Cerritos, California, learned to program in Basic on the IBM Personal Computer. Angela is twelve, Matt is "twelve going on thirteen," and Carlene is thirty-seven. All three were participants in Microcomputers in Education, an intensive hands-on workshop that took place in June and July on the campus of Stanford University in Palo Alto, California.

Here, thirty-six highly motivated educators and sixty-six young people from the Computer Tutors at Stanford day camp gathered to find out more about computers and to develop the skills needed to work with computers successfully.

The six-week institute and five-week summer camp were cosponsored by the School of Education at Stanford University and Interactive Sciences, a nonprofit corporation based in Palo Alto. Funded primarily by the Luke B. Hancock Foundation and the Walter S. Johnson Foundation, Interactive Sciences has for several years been developing new ways to teach computer literacy to both children and adults. The IBM Personal Computer systems used for instruction were loaned by IBM.

Global Response. Educators who attended Microcomputers in Education came from as close by as San Jose and Oakland, California, and as far away as Chicago, Illinois, and Geneva, Switzerland. Some had worked with or around microcomputers before, but the majority had little experience actually using them.

During the first week, the educators got fifteen hours of computer orientation and fifteen hours of programming instruction. By the time the end of week two rolled around, these same educators were confidently guiding junior high school-age computer campers in the creation of elementary Basic programs. Later on, the educators would learn and teach word processing and would explore such areas as graphics, robotics, music, and turtle graphics on their own and with their students. They would also learn and teach intermediate-level Basic programming.

How can people who only a week before may not have known the difference between *rem* and *run* teach Basic to a new group of learners? It may sound strange at first, but situations in which people who have just learned something themselves instruct another group of beginners are fairly common occurrences in the San Francisco Bay Area these days—and once skeptics see the process in action, they're hard pressed to challenge the arrangement.

Central to the success of the interconnected institute and day camp projects is a teaching method known as peer tutoring. By now, in the Bay Area at least, the technique is closely associated with the Interactive Sciences organization and with the more than fifteen hundred elementary and secondary school students in the Palo Alto School District over the last three years who have earned the right to be called Computer Tutors.

Preteen Tutors. Computer Tutors are people, many of them as young as ten or eleven, who have completed at least six lessons in elementary Basic programming and who agree to teach the lessons they have mastered to two new students. The tutors learn by teaching, and the new students get the kind of individual attention not

Inset, left to right, program creators Joan Targ, Jeff Levinsky, and Decker Walker.

available in traditional classroom settings.

During their first week in the program, prior to the campers' arrival, teachers and administrators were taught by young Computer Tutors from the Palo Alto School District. And at the end of their own work with the campers, the educators would qualify to be Computer Tutors themselves.

The first Computer Tutor sites were established in the Palo Alto School District in 1979 by Joan Targ, funded initially by a federal Title IV-c grant. A teacher at Jordan Middle School in Palo Alto, Targ is also the president of Interactive Sciences. Confident in the power of teaching as a way of learning, Targ, with Interactive Sciences's vice president Jeff Levinsky, a computer scientist, has been expanding the Computer Tutor network ever since. Studies have shown that students learn much more from their experiences tutoring others than they would in more traditional setups.

Educators in this summer's Stanford program worked from *Basic Basic for the IBM Personal Computer*, a student manual prepared especially for them by Targ and others. The typical arrangement of the tutoring situation was for one tutor to spend three hours every weekday working with two students who shared a computer. Students were encouraged to work at their own pace.

At the program's beginning, teachers and school administrators were taught by Computer Tutors; at various intervals later on, the teachers assumed the instructor role with the campers, spreading the wealth and reinforcing their own new knowledge.

Throughout the program, learners of all ages were assisted by instructors Targ, Levinsky, and Decker Walker. Walker is Associate Professor of Education at Stanford's School of Education and codirector of the Interactive Educational Technology master's degree program. Also on hand were the circulators—advanced level Computer Tutors or students in the IET program who know how to help if the going gets rough. The Stanford students also work as camp counselors in the afternoons.

Diving Right In. Participants studied computer vocabulary, got acquainted with the IBM Personal Computer's inner workings, learned to make flow charts of programs they wanted to write, and discovered they could indeed make sense out of *if-thens*, *gotos*, and *for-next* loops. Educators also acquired a working knowledge of CAI, word processing, and computer-assisted school management techniques.

Targ explains that most people are hesitant at first about the idea of teaching. "What, me teach?" they say. "I've just learned."

"Yes," says Targ, very much at ease with the idea. She is thoroughly convinced that anyone can learn about computers and believes that one very good way of learning is by teaching someone else what you know so far. Even the computer campers (who are taught by the educators who learn from the Computer Tutors) learn to teach others how to use computers; that way, the youngsters can share their new expertise with family and friends and the chain can continue.

Targ has been teaching people about computers for the last eight years. During that time, she has seen that people's attitudes toward themselves and toward computers dramatically affect their ability to learn. Therefore, one of the primary objectives of the workshop coordinators was to help participants feel comfortable with computers. The leaders worked to provide a supportive environment in which participants could discover their own capacity to understand and make use of computer technology.

Luxury Learning. It was evident to a visitor who dropped in on the program in early July that the computer campers were feeling at ease in and thoroughly enjoying the rich learning environment. Perhaps Terrell Prude, a thirteen-year-old from Portola Valley, California, was the most straightforward in expressing his feelings of self-confidence and enthusiasm.

"Programming is easy; it's a cinch," Prude declared. He was talking about the exercise he and learning partner Soren Juul, eleven, were working on with assistance from San Jose educator Meg Self. Prude and Juul then proceeded to explain why *rem* statements are used in programs and why you must put something in quotes if it is part of a *print* statement. They also spoke from experience about syntax error, that notorious nemesis of programmers everywhere, with which they were by now all too well acquainted.

The boys were in the middle of writing a program called *Pizza Pie.* The program required finding the diameter of a circle in order to determine the price of pizza per square inch. Self, who had completed the same exercise just a week earlier, explained that when the boys got this program to run correctly it would demonstrate that they had mastered the rules of precedence governing arithmetic operations. Clearly, just as Self's experience helped her students learn better, watching them work through the exercise helped cement Self's own understanding of the material.

Meanwhile, at another computer station, Pandora Bednar of Brisbane, California, and Kim Farey of Menlo Park were busy writing *Love Letter,* a program that required them to use the Basic commands they'd learned earlier. Their tutor was Dr. Evelyn Wesley, a vocational specialist from Oakland, California. The easy rapport between students and teacher helped create a relaxed atmosphere in which it was safe to question, show enthusiasm, and share accomplishments.

The Best for Last. Interactive Sciences has a particular interest in seeing that people who tend to get left behind when it comes to technology—women, seniors, minorities, and the handicapped—have the opportunity to develop computer awareness and skills. They try to counterbalance stereotypes by providing positive role models.

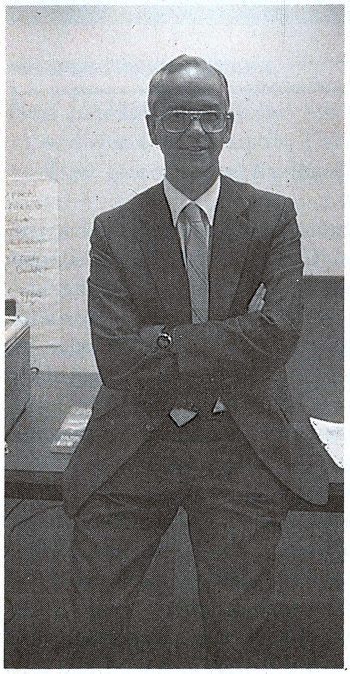
In the Stanford summer program and in the Computer Tutor project in general, importance is placed on pairing girls and women with tutors of the same sex. While boys and men seem to do equally well learning from male or female teachers, strong female role models seem to make a real difference in the way girls and women learn. Girls and women identify with a female tutor; when they see how capable she is, and how much she enjoys what she's doing, they tend to feel more confident that they, too, can be successful with computers.

Word processing on the computer is another confidence builder. Joan Targ remarks that students who have had trouble with the mechanics of putting together acceptable assignments usually love working at the computer, and their language arts skills improve significantly as a result of the process.

Targ describes Microcomputers in Education as "the first effort of its kind," but promises that workshops and consultations to other schools are in the planning stages. Curriculum materials for teaching advanced Basic, Pascal, and assembly language are also being developed. When ready, these materials, along with a comprehensive management manual, will be made available to schools that want to set up centers of their own.

Grand Design. Setting up centers in their schools or districts is just what educators who attended this summer's workshop hoped to do upon returning home. To establish and operate their own centers, educators needed new knowledge, new skills, and new experiences. In organizing Microcomputers in Education, Targ, Levinsky, and Walker made an effort to provide such opportunities.

The result was an impressive combination of individualized instruction, computer orientation, exposure to computer technology (via field trips, speakers, and demonstrations), and hands-on expe-



Coaching Teachers

The guest speaker on Friday, July 2, was Bob Rowe of IBM. Rowe has degrees in both physics and law but has spent the last nineteen years in marketing at IBM; at present, his job is the educational marketing of the IBM Personal Computer.

According to Rowe, IBM sees small businesses, educational settings, and homes as the main places the IBM Personal Computer will be used. Loaning more than thirty systems to the Stanford project was a way for IBM to demonstrate its belief in the importance of teacher training and the development of computer centers.

In the course of the afternoon, Rowe gave his listeners an overview of the IBM Personal Computer system specifications, provided information about software for the pc, and explained how to take advantage of a 15 percent educational allowance on the purchase of twenty or more Personal Computers. He said that the focus of new developments in computing "tends to be more on speed and function than on coming down further on price."

Some Gripes, Too. Rowe also discussed graphics and communications capabilities, fielded questions, and listened to gripes—the chief of which concerned the placement of the shift key on the IBM keyboard. Several educators found it far from user friendly. The scales seemed to come back into balance when someone else made the point that the detachable keyboard is "good in a tutoring situation because you can pass it around." Everyone agreed about that, and about the wisdom underlying the "intentional awkwardness" of the control-alt-delete maneuver. All in all, the Personal Computer keyboard got high marks from the educators.

Early in his presentation, Rowe traced IBM's history of involvement with computers for education, beginning with Coursewriter 1 (an authoring language released in the late fifties that enabled people to write their own courses). But he wound up devoting much more time to a discussion of the present and future in educational computing.

According to Rowe, "computer literacy is the most widespread application" in education right now; drill and practice is next, followed by computer-managed instruction. In his view, "the world doesn't know very well yet how to use computers for instruction."

Rowe also addressed the pros and cons involved in generating one's own courseware. He stressed that "authoring processes are complex," involving logic, branching, the capability to anticipate answers, and so on. While developing one's own modules can be fairly easy, "producing quality courseware is a time-consuming process."

Rowe expressed the hope that good simulation programs will be among the software we see in the future. He suggested simulation of difficult, dangerous, or time-consuming scientific experiments as one example of the creative and useful functions educational software could serve in the future. Vocabulary drills, tutorials, career

rience in running a peer-tutoring computer center.

It's likely that educators who completed this program will feel well-equipped—confident to move ahead on their own. Interactive Sciences also plans to follow up on the activities and progress of attendees in order to evaluate the effectiveness of the program.

Targ and Levinsky see the wisdom of involving everyone concerned—administrators, teachers, parents, and kids—in the learning process. They've found that when people are encouraged to participate in an activity—asked to help make it happen rather than being told how things are going to be—they bring enthusiasm and ideas that shape the experience and make it better. People feel good about themselves and help create an atmosphere of optimism in which many good things are possible.

Running with It. It's not surprising that the esprit de corps and optimism of those involved are among the most striking aspects of the Microcomputers in Education program. Attendees and instructors alike seem to take pride in what they are doing, enjoy the sense of accomplishment that learning gives them, and welcome new challenges. Computer Tutor T-shirts, such as the one worn by Joan Targ, and the banners and certificates that adorn the walls of the computer learning center reinforce these feelings of camaraderie and competence.

Fittingly, the certificates displayed on the wall are awarded to those who have completed the first series of lessons in Basic programming. And even more fittingly, the certificates are made by the students themselves. Making one's own certificate is the very last exercise in the *Basic Manual*; in order to complete it successfully, students must put their newly acquired formatting knowledge to good use.

It's not hard to understand why these handcrafted certificates mean more to participants than would an award created by someone else. After all, being able to make things happen yourself is a great feeling. ▲

guidance applications, and computer-managed instruction (testing, prescribing instruction, and tracking progress) were also mentioned as areas for future educational software.

Rowe had some recommendations for educators attending the Microcomputers in Education program. "You are going to find yourself in the position of having to defend the use of computers in education," he reminded his listeners. "Make a list for yourself now of what you want to walk out of this workshop with. Start talking to your cohorts now about what you need here and how to get it."

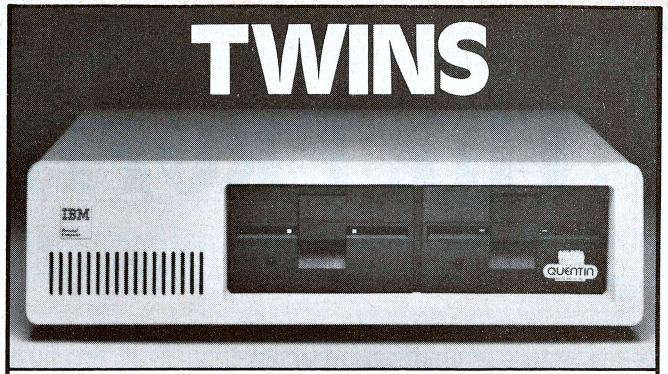
Brass Tacks. Rowe also noted the kinds of practical considerations that educators would need to take into account when actually setting up a center, including objectives, resources, school budget cycles, staffing, hardware/software selection, teacher training, validation, and so on.

He expressed the belief that information networks will play a large part in the future of education, especially now that we are beginning to make computers easier for people to use. He suggested that we are moving into a time when "reading, writing, and computing" are coming to be seen as basic skills, a time when the computer may become a bridge between the home and the school.

"Over the next five years," Rowe said, "parents will be given a tool that will enable them to be much more involved in their children's education. Computers can serve as a trigger to help parents realize that they can influence their children's education," and the schools will need to be ready to deal with this development.

Rowe sees the computer facilitating our movement into a time when discovery and meaningful learning can take place more easily, without negativism and traditional concepts of failure. He hopes that we will become more and more comfortable with the idea of lifelong learning.

In Rowe's view, and that of IBM, we are taking "very small steps toward very great things." ▲



Disk Drive Add-Ins From Quentin

Team up with Twin-Set™ from Quentin and go for the extra points in disk drive capabilities. Twice the storage, faster access, and Twin-Set's high quality Lead Screw Positioner will put your IBM PC ahead of the game.

Twin-Set is fully software transparent with IBM DOS and CP/M, and easily slides into the basic chassis. It is built around a reliable Siemens system with over 10,000 lifetime hours. MTBF 8500 power-on hours. On Track Head Seek virtually eliminates retries and copying errors, and diskette protection system guards disk hubs.

You waited for IBM...now add the components that will give you top performance. Special introductory price $249.00. Full one-year factory warranty. Count on Quentin for Quality... in disk drive systems.

To order: Check, Visa or Mastercard number. Calif. residents add sales tax. Call for specifications. Club discounts available.

QUENTIN RESEARCH, INC.

QUENTIN RESEARCH, INC.
19355 Business Center Dr.
Northridge, CA 91324
(213) 701-1006
Telex: 910-493-2174

Winchester Disk Drive From Quentin

Faster than a speeding floppy. More powerful than a dozen diskettes. Able to store and retrieve 5, 10, 15, even 20 megabytes of formatted data on a single drive.

It's Big Brother, the IBM PC Winchester Disk Drive from Quentin. **Fast.** Accelerates from 0 to Read in 70 MS. DMA data transfer rate zips along at 5 MBits per second. **Powerful.** Compatible with IBM DOS, CP/M and protected software. 20 megabytes equivalent to 140 floppies. **Reliable.** Flick a switch and Big Brother's Self Test Diagnostic system checks all Winchester operating functions, verifying the integrity of the drive and the data. 32 bit Error Correction Code overrides all data field errors. MTBF 10,000 power-on hours, with no preventive maintenance required. Full one-year factory warranty. **Affordable.** 5MB $1995. 10 MB $2195. 15 MB $2395. 20 MB $2595. Count on Quentin for Quality... in mass storage subsystems.

To order: Check, Visa or Mastercard number. Calif. residents add sales tax. Call for specifications. Club discounts available.

QUENTIN RESEARCH, INC.

QUENTIN RESEARCH, INC.
19355 Business Center Dr.
Northridge, CA 91324
(213) 701-1006
Telex: 910-493-2174

THE PROFIT PLOT

by Shirley Fisher

Does your company have six dozen widgets sitting around in inventory? Have they been there since the year 1843? Are you out of satchel whistles again? Have you just discovered you promised three hundred stork hats to be delivered in four working days when your supply on hand is down to twenty and you can't get more for at least a month because the Feather Pluckers Union is on strike?

In one case you've got money and space tied up in a slow moving (maybe even dead) product. In another, good relations with a preferred client may be jeopardized. What can you do to improve your situation?

Keeping track of your current inventory position can be very costly—in both time and money. A manual method can take several people several days. By the time they've completed the task, the actual inventory may have changed. That's okay for a once a year tax-time chunk of information gathering, but when it comes to maintaining tight control of your assets, cash flow, customer good will, and so on, a system that gives you more frequent and current information would be far more valuable.

An inventory is a necessary part of doing business for most companies. Even blood banks have to keep track of how much stock they have on hand. How do you select the best inventory policy for your firm? When do you replace inventory items? How much is enough? Too much?

Your computer can be a big help in maintaining inventory records and signaling when and how much to reorder. You can use it to formulate a mathematical model that describes the behavior of your inventory system.

Some of the costs that determine the profitability of your inventory model are: costs of ordering or manufacturing; storage costs (or holding); shortages (costs of unsatisfied demands); revenues; salvage costs; and discount rates.

We'll look at a couple of inventory models and discuss their implementation with electronic spreadsheet programs. Several of the formulas we'll use involve differential calculus; we'll omit some details concerning their derivation and simply present mathematical expressions that you can plug into your own models.

Set 'Em Up. Whether you are manufacturing a product or ordering it from another company, there are certain costs, called set-up costs, which we will represent by the letter K. These would include such things as administrative costs of ordering, preliminary labor, and other expenses related to starting up a production run.

We'll let the letter C represent the production cost per item. The letter H will stand for the holding or storage costs associated with keeping the inventory until it's used or sold. Holding costs could include the cost of capital, warehouse space, insurance, a security guard, and taxes on inventory. This cost can be calculated on the basis of the maximum quantity held during a period, the average quantity held, or the cumulated excess of supply over demand.

Caught Short. Since you can't sell what you don't have, shortages also have their related costs. These costs depend on the structure of your model. When a demand exceeds the available inventory, it can be met by a priority shipment or not met at all. If it's met on a priority basis, the cost of the shortage may be the difference between special handling and ordinary shipment. If the demand isn't met at all, the cost is the loss of revenue and, possibly, of the customer's good will.

Revenue cost may or may not be included in your inventory model. If price and demand are not under your company's control, your inventory policy is not dependent on sales revenue. Loss in revenue, however, is included in the unsatisfied demand penalty cost whenever you can't meet demand and a sale is lost. Even the cost of delay can be included in the unsatisfied demand cost.

Salvage value is the worth of leftover items at the end of the inventory period. If the inventory policy is carried on indefinitely and there is no obsolescence, there aren't any leftover items. What's left over at the end of one period is the amount that begins the next period. The salvage value sometimes represents the disposal value, or selling price. The negative of salvage value is salvage cost. If there are costs connected with an item's disposal, the salvage cost may be positive. Salvage costs and storage costs are usually combined as one item.

The discount rate is a measure of the time value of money. When you have capital tied up in inventory, you don't have this money available to invest in other areas, so there's a cost associated with it. If you invested this money in government bonds at 12 percent, your return in a year would be $1.12 for every dollar invested.

Demanding Models. Inventory models can be categorized several ways. Sometimes they're classified according to whether the demand for a period is known (deterministic demand) or is a random variable with a known probability distribution (nondeterministic, or random, demand). Time lags in delivery are frequently characteristic of random-demand models.

Another type of model classification relates to whether the inventory is reviewed continuously or periodically. In a continuous-review model, as soon as the stock level falls below a preset reorder point, an order is placed to replenish the item. In the case of a periodic model, the inventory is checked at definite time periods, such as monthly—and reorder decisions are made only at these checkpoints. If the interval between period reviews is small enough, the periodic model can be used to approximate a continuous review policy.

We'll look at an inventory problem where the actual demand is known. This particular model is sometimes called the economic lot-size model. Items in this model are withdrawn continuously at a known constant rate over a specific period of time—say, one month. This is a common situation in a manufacturing environment where a particular item is a part included in an assembly-line production item.

Scarlet Letter. We'll identify the constant rate in our model by the letter A. For simplicity, we'll assume that the item in question arrives in lots of Q quantity with no delay involved. The only other costs we need to consider are set-up cost, K; product cost per item, C; and inventory-holding cost per item per month, H. Lines 7 to 11 of figure 4 summarize our definitions so far.

What we're trying to find out from this model is how often we should make a production run and what size run will minimize the cost per unit of time. This is a type of continuous-review model in which we'll assume also that no shortages are allowed.

A cycle is the time between production runs and can easily be represented graphically (see figure 1). Let time, t, be represented on the x axis, and let the inventory level, Q, be represented on the y axis. At the

beginning of each time period, you have just received your entire order of parts for the period—say, 24,000. At the end of the period you have just used up your last part and have zero parts left over. Your supply of parts can be represented by a line drawn from the point on the y axis that represents 100 percent of Q (at the beginning of the period) to the point on the x axis (at the end of the period) where y is zero. Your graph will look like a right triangle.

If you graph more than one time period, you will have a series of adjacent right triangles along the x axis. The point on the x axis where the first two triangles meet is Q/A, the cycle length. The second meeting point is 2Q/A, and so on.

Productive Cycle. The production cost per cycle is given by the expression K+CQ, where Q is greater than zero. The holding cost per cycle is the holding cost per unit multiplied by the average inventory level during the cycle (HQ/2) multiplied by the cycle length (Q/A). With these two costs we can calculate the total cost per cycle, which is the production cost per cycle plus the holding cost per cycle:

$$\text{Total cost per cycle} = K+CQ+(HQ/2)(Q/A)$$

From this information we can calculate the total cost per unit of time, T, which is the total cost per cycle divided by the cycle length.

Now let's take an example situation and see what the model can tell us.

A computer manufacturing company makes its own keyboards, which are used in the production of its line of personal computers. The computers are assembled on a continuous production line at the rate of 8,000 a month. The keyboards are manufactured in batches because large quantities of them can be produced in a fairly short time. The company is trying to determine how often to schedule a production run and how many keyboards to produce per batch.

For each run, a set-up cost of $12,000 is incurred. This includes the cost of setting up the production line (called tooling up), record keeping, and administrative costs—among other things. Because the set-up cost is a sizable item, the company management may opt to produce keyboards in very large batches. On the other hand, this can lead to a large inventory.

Choose Your Weapons. The cost of keeping a keyboard in inventory for one month is $.30. This cost includes storage space, capital tied up, insurance, taxes, and so on. Management may wish to hold this cost down by producing keyboards in smaller batches to reduce inventory levels.

To keep our model from becoming too complicated, we'll assume that the production cost of a single keyboard (excluding the set-up cost) is $10, regardless of the batch size. In a real business situation, however, this usually will not be the case, as the per-unit cost generally decreases with increasing batch size.

In this model, we are not allowing the company to plan any shortage of parts, but if a shortage of keyboards does occur, there will be a per-unit cost, P, of $1.10 to cover the cost of storing the incomplete computer, installing the keyboard after the computer is fully assembled, delayed revenue, and extra record keeping.

To summarize, we have the following initial values:

A = 8000
K = 12000
H = 0.30
C = 10.00
Q = 24000
P = 1.10

We'll call the optimum quantity, that value of Q that minimizes T (the total cost per unit of time), Q'. With a little calculus we can deduce that the value of Q' is such that $-AK/QQ + H/2 = 0$. Or, in other words, Q' is equal to the square root of 2AK/H.

The Clock Is Ticking. The other value we're interested in is the time it takes, T', to use up the optimum quantity, Q'. This value can be found by dividing Q' by A.

By plugging our initial values into these two formulas, you should find that Q' is 25,298 keyboards, and T' is 3.16 months.

By making a slight variation in the model, we can see if it would be profitable to allow shortages of keyboards to occur. The cycle length may be increased, permitting a savings in set-up costs. On the other hand, this benefit may be offset by the costs incurred when there are shortages.

The proper solution is not intuitively obvious. Figure 2 is a graph similar to that of the no-shortage model (figure 1). In the first graph, the quantity we needed was equal to the stock on hand. This time the stock on hand is less than the quantity needed. The quantity S in figure 2 represents the stock on hand, while Q represents the quantity of keyboards needed.

In figure 2 we have two right triangles

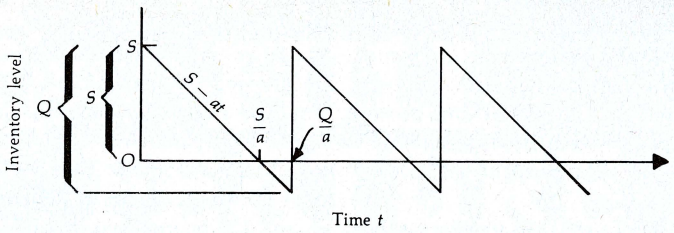


Figure 2. Inventory level as a function of time—shortages permitted.

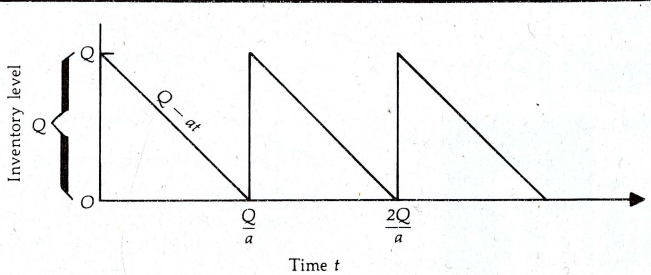


Figure 1. Inventory level as a function of time—no shortages permitted.

International Computers

IBM PC Software Available NOW

Mister Lister--"Space Shuttle of Mailing List Programs"--Sort over 2,000 names in one minute! **$79.95**

SerPrinSoft--Does your Serial Printer run at full Baud Rate?--If not, you need SerPrinSoft. Hardware & XON-XOFF protocol supported. **$79.95**

Go-Forth--Know Forth in 2 hours? Well maybe not, but certainly get acquainted with the help of Computer Assisted Instruction. Forth language & editor included. **$59.95**

Speedup-Plus--Faster than Speedup--Virtual memory for the IBM. **$199.95**

79-Standard Fast Forth--Fast Forth® is faster than any micro-computer Forth. **$499.95**

All programs include the Forth language. Minimum system: 1 disk drive & 64K.

Phone or Write:

I.C.
13048 Olive Blvd.
St. Louis, MO 63141
(314)878-8838

per cycle, one above and one below the x axis. The one above represents the portion of the time period in which stock is available, and the one below represents the portion of the period when stock has run out—the shortage period. The fraction of time during which no shortage exists is represented by P/(P+H) and is independent of set-up costs, K.

The Finishing Touch. We need one other intermediate value in order to calculate our shortage model. That is (P+H)/P. There are two decision variables in this model, S and Q; so the optimum values S* and Q* are found by setting the partial derivatives of T with respect to S, and T with respect to Q, equal to zero.

If we clean up the calculus and solve the equations simultaneously, we find that S* is equal to the square root of 2AK/H times the square root of P/(P+H). Since the first of these square roots is equal to Q′ (according to the equation we developed earlier), S* turns out to be Q′ times the square root of P/(P+H). The optimum period length, T*, becomes Q*/A, and the maximum shortage is given by Q* − S*. If our arithmetic is correct, we now should have 22,424 keyboards for S*, 28,540 keyboards for Q*, 3.57 months for T*, and 6,116 keyboards for the maximum shortage, Q* − S*.

Now that we've got the numbers, what do they mean?

Well, if no shortages are allowed, we should set up the production line every 3.16 months and produce 25,298 keyboards. If we generate a cost curve, we'll see that production quantities between 20,000 and 30,000 are nearly as acceptable. If shortages are allowed, we should set up the production line every 3.57 months and produce 28,540 keyboards. We can allow a shortage of 6,116 keyboards. You will notice that there isn't a whole lot of difference between Q′ and Q* or between T′ and T*.

The models we've looked at have assumed that the unit cost of the keyboards was the same regardless of the quantity produced. A result of this assumption is that the optimal solution is independent of the unit cost. It's much more common to expect a price break on larger quantities, so let's take a brief look at that situation.

The Bigger the Better? Let's assume that if we produce fewer than 10,000 keyboards per run, the cost per item, C, will be $11 instead of $10. If we produce between 10,000 and 80,000 keyboards, the production cost is $10, and if we produce more than 80,000 units, the cost drops to $9.50 per keyboard. If this were the situation, what would be the best inventory policy, assuming no shortages are allowed?

To deal with that question, we can use an electronic spreadsheet to help us calculate some data points—and then use that information to plot a graph of the three unit price curves, plotting quantity on the x axis and total cost per unit of time on the y axis. Such a plot is shown in figure 3; notice that the shape of the curve is the same for all three unit prices. We are interested in only a portion of each of the curves.

From the economic lot-size model, the formula that defines each curve in figure 3 is:

$$T(j) = AK/Q + HQ/2 + AC(j)$$

for j=1,2,3.

By plotting the three price curves together, we can easily see that the optimum value of Q, the value that minimizes T(j), turns out to be 25,298 keyboards. This number is not at all feasible for the 10,000-piece lot, so we can eliminate that possibility from our policy considerations. Let's take a look at the $9.50 versus $10 cost curves.

Throwing a Curve. The lowest portion of the $10 curve in the quantity region of 80,000 and above occurs at 80,000 units, at a cost of $93,200. We've already discovered that the optimum value of Q is around 25,000 units. By plotting some quantities on both sides of this value, say 20,000 to 40,000 units, we can see that with 20,000 units at $10, T is $87,800; with 30,000 units, $87,700; and with 40,000 units, $88,400. All three of these values are less than that of 80,000 units at $9.50, which is $82,200. If, however, the quantity discount brought the per-unit cost down from $9.50 to $9, then the optimal production quantity would become 80,000 and the total cost per unit of time would be $85,200.

There are several things we can observe from economic lot-size models:

1. If the production cost of an item is constant over time, it doesn't affect the optimal solution. In other words, no matter what policy is used, the same quantity is the optimum quantity, so this cost is a fixed cost.

2. We assumed that Q, the number of keyboards produced, was constant from cycle to cycle. This is actually a result of the model, rather than an assumption.

3. The economic lot-size model can be looked upon as a special case of an (s,S) policy used as a periodic review policy (where, at review time, an order is placed to bring the inventory level up to S if the current inventory is less than or equal to s; otherwise, no order is placed). The symbol S represents the reorder level; s denotes the reorder point.

In an economic lot-size model, s represents the level at which items are ordered. When shortages are not allowed, the reorder point is zero. When shortages are allowed, s represents the negative of the maximum shortage, or −(Q* −S*). Since the economic lot-size model is continuously reviewed, an order is placed when the inventory level equals s, which brings the inventory up to reorder level S. In the (s,S) policy, this would tell us, when the inventory level reaches the reorder point, s, to place an order to bring the inventory level up to the reorder level, S.

4. The reorder point will never be a positive value. If shortages are allowed, s will be less than zero, and if shortages are not allowed, s will be zero.

5. A known, fixed lag time can be cal-

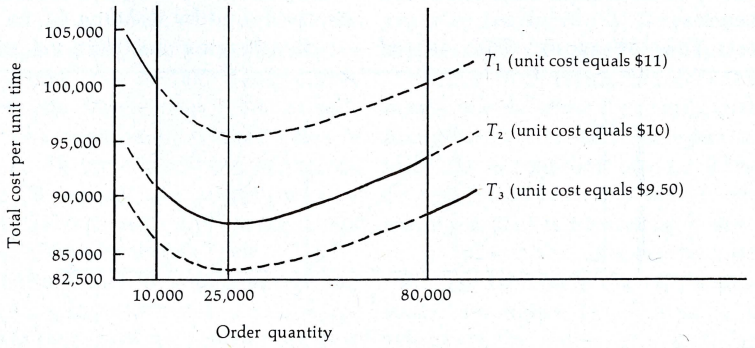


Figure 3.

WABASH MINI-MYTE

Certified Diskettes

1 Year Perf. Warranty
Hub Rings, Write Protect
Stickers, Labels, Heavy
Jackets, Envelopes and Box

Computer	1-9 Boxes	Multiples of 10 Boxes
Apple II	$21.95/Box	$20.95/Box
TRS-80 MOD I/III	$23.95/Box	$22.95/Box
IBM Personal	$23.95/Box	$22.95/Box
TRS-80 MOD II	$33.95/Box	$31.95/Box

Others available on request.
Send Certified Check or Money Order
If personal check allow 2 weeks
Add $1.50 Post & Handling per order
Illinois Residents add Sales Tax

Send to:

Com-Pute Services
P.O. Box 3361
Oak Park, IL 60303

culated into the model. Let L be the lead time required between placing and receiving an order. Assume that L is constant over time and not dependent on the size of the order. Then if the order is to arrive the moment that inventory level reaches s, the order must be placed L periods ahead of time. So when the lead time multiplied by A (the number of units required per unit of time) is less than Q (the number of units ordered at a time), the recorder point is s − LA.

In figure 4 you'll see a summary of the symbols we used for the economic lot-size model, as well as a short definition of each symbol. The way in which you lay out your electronic sheet is entirely up to you. We have grouped together the simple variables in one area of the sheet in order to simplify our "what if" use of the model.

One from Column A. Under the labels NO SHORTAGES and SHORTAGES we have grouped the intermediate values, followed by the items of primary interest—TC/CL, Q′, T′, S*, Q*, T*, and Q* − S*. Beneath all of this, we have set up an additional set of initial values of C and Q for use in plotting the curves shown in figure 3. Whether you use *VisiCalc* or *SuperCalc* or another electronic spreadsheet, you will still have to calculate the same basic equations.

Initial values and their coordinate locations are:

A = B29
Q = B30
K = B31
H = D29
C = D30
P = D31

Intermediate values and values of primary interest were as follows:

K+CQ, at B35:	+B31+(D30*B30)
Q/A, at B36:	+B30/B29
HQ/2, at B37:	(D29*B30)/2
HQQ/2A, at B38:	+B36*B37
PC+HC, at B40:	+B35+B38
TC/CL, at B41:	+B40/B36
P/(P+H), at D35:	+D31/(D31+D29)
(P+H)/P, at D36:	(D31+D29)/D31
Q′ at B43:	@SQRT((2*B29*B31)/D29)
T′, at B44:	+B43/B29
S*, at D42:	+B43*(@SQRT(D35))
Q*, at D43:	+B43*(@SQRT(D36))
T*, at D44:	+D43/B29
Q*−S*, at D45:	+D43−D42

Additional values used to plot points for our quantity discount curves were entered at B47 through B50 for C1 through C4, and at D47 through D50 for Q1 through Q4. In column B, rows 54 to 57, we used our initial value for Q while we let the cost, C, range

JUST $365

FULL FEATURE ACCOUNTING

apple IBM NEC HEWLETT PACKARD

The INSOFT ACCOUNTANT is a professional menu-driven accounting system organized into 4 packages — General Ledger, Accounts Receivable, Accounts Payable and Payroll. Each package may be used independently or will post directly to the General Ledger. Designed for easy startup, the INSOFT ACCOUNTANT is supplied with comprehensive training manuals and complete sample data files for training and practice before actual installation. The price of $365 includes all packages, sample data files, and manuals handsomely packaged in individual binders.

GENERAL LEDGER

Balance Sheet:	Summary — Shows current and YTD values
	Comparative — Above plus budget and last year
Income Statement:	Summary — Shows current, YTD values and %
	Comparative — Above plus budget, last year, month and YTD
	Dept. Income Statements, comparative and summary
Trial Balance:	Complete transaction detail by account number

ACCOUNTS RECEIVABLE

Invoice:	Quick single billing or batch inv. entry; allows discounts
Statement:	Automatic billing; invoices automatically entered
Aged Receivables:	Detail or summary reports with 4 aging periods
Customer Activity:	Reports to video screen or printer

ACCOUNTS PAYABLE

Checks:	Uses commercially available forms; allows automatic payments
Cash Requirements:	By due or discount date; excellent cash management tool
Check Register:	Check Record for general use and audit
Query Vendor:	Account status by vendor to video screen or printer
Aged Payables:	5 Aging periods

PAYROLL

Payroll Checks:	Comprehensive check stub, uses commercially available forms
941A:	Prints end of quarter report values
W-2:	Prints end of year forms

PRICE BREAK THROUGH JUST $365

SYSTEM REQUIREMENTS

Apple:	Apple II, Z-80 Softcard, 56K
Hewlett Packard:	HP 125
IBM:	IBM Personal Computer
NEC:	PC-8000 Series
8" Disk:	Standard single density format

In addition, all versions require 2-disk drives or hard disk and a 132-col. printer.

UPGRADE YOUR BUSINESS ACCOUNTING.

Send for our free brochure and sample printouts

insoft®

10175 S.W. Barbur Blvd. Suite 202B
Portland, OR 97219

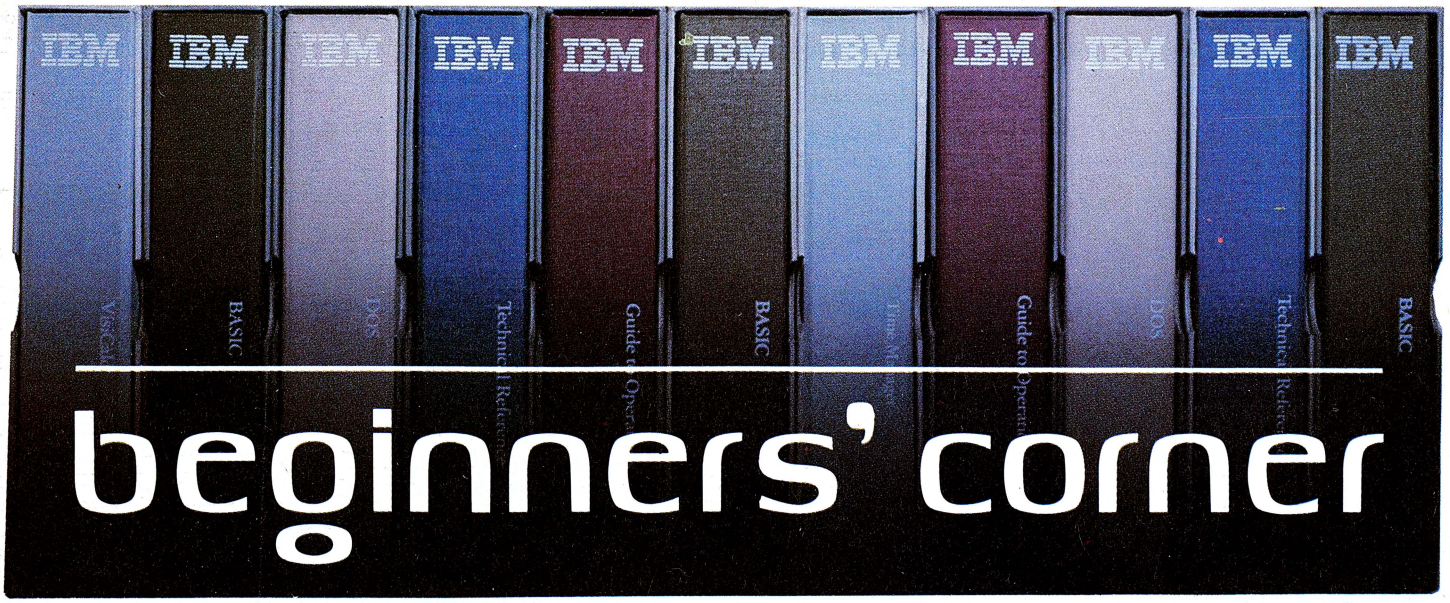
(503) 244-4181

Apple, CP/M®, and MBASIC, are registered trademarks of Apple Computer Inc., Digital Research Inc. and Microsoft, respectively.

from $11 to $9. In column D, rows 54 to 57, we used Q1 through Q4 for the Q value:
T(1), at B54:
(B29*B31/B30)+(B30*D29/2)+(B29*B47)
T(2), at B55:
(B29*B31/B30)+(B30*D29/2)+(B29*B48)
T(3), at B56:
(B29*B31/B30)+(B30*D29/2)+(B29*B49)
T(4), at B57:
(B29*B31/B30)+(B30*D29/2)+(B29*B50)
with Q1, at D54:
(B29*B31/D47)+(D47*D29/2)+(B29*B47)
with Q2, at D55:
(B29*B31/D48)+(D47*D29/2)+(B29*B48)
with Q3, at D56:
(B29*B31/D49)+(D47*D29/2)+(B29*B49)
with Q4, at D57:
(B29*B31/D50)+(D47*D29/2)+(B29*B50)

	A	B	C	D	E	F	G
1							
2							
3							
4	DEFINITIONS FOR TWO SIMPLE DETERMINISTIC MODELS — INVENTORY						
5							
6							
7	A = # OF UNITS REQUIRED PER UNIT OF TIME(MONTH)						
8	Q = # OF UNITS PRODUCED AT A TIME						
9	K = SET-UP COSTS OF PRODUCTION						
10	C = PRODUCTION COST PER ITEM						
11	H = INVENTORY HOLDING COST PER ITEM PER UNIT OF TIME						
12							
13	P = $ COST PER UNIT OF UNFILLED DEMAND PER UNIT OF TIME						
14	S* = MIN STOCK IN A CYCLE						
15	Q*−S* = MAX STOCK SHORTAGE IN A CYCLE						
16							
17	K+CQ = PRODUCTION COST PER CYCLE						
18	Q/A = CYCLE LENGTH						
19	HQ/2 = HOLDING COST						
20	(HQ/2)*(Q/A) = HOLDING COST PER CYCLE						
21							
22	TOTAL COST PER CYCLE = PRODUCTION COST + HOLDING COST						
23	TC = PC + HC = (K+Q) +((HQ/2)*(Q/A))						
24	T = TOTAL COST PER UNIT OF TIME						
25	= TOTAL COST PER CYCLE/CYCLE LENGTH = TC / CL						
26							
27	INITIAL VALUES:						
28							
29	A =	8000	H =	0.30			
30	Q =	80000	C =	10.00			
31	K =	12000	P =	1.10			
32							
33	NO SHORTAGES		SHORTAGES				
34							
35	K+CQ =	252000	P/(P+H)=	.7857143	= TIME NO SHORT EXISTS		
36	Q/A =	3	(P+H)/P=	1.272727			
37	HQ/2 =	3600					
38	HQQ/2A =	10800					
39							
40	PC + HC =	262800					
41	TC / CL =	87600					
42			S* =	22424			
43	Q′ =	25298	Q* =	28540			
44	T′ =	3.16	T* =	3.57			
45			Q*−S* =	6116			
46							
47	C1 =	11.00	Q1 =	10000			
48	C2 =	10.00	Q2 =	80000			
49	C3 =	9.50	Q3 =	80000			
50	C4 =	9.00	Q4 =	80000			
51							
52	T(J) = AK/Q + HQ/2 + AC(J)						
53							
54	T(1) =	101200	WITH Q1=	99100			
55	T(2) =	93200	WITH Q2=	93200			
56	T(3) =	89200	WITH Q3=	89200			
57	T(4) =	85200	WITH Q4=	85200			

Figure 4.



beginners' corner

by Craig Stinson

If you've been reading this series from the beginning (June 1982), you'll remember that we started with some hands-on exercising. You loaded the Disk Operating System, typed *BasicA Samples* to run the demonstration programs that IBM supplied on your DOS disk, and then returned to the operating system to make a backup copy of DOS.

Last month we started retracing those steps and examining in more detail what was going on inside your computer at the time you executed them. We had arrived at the point where you had finished loading DOS into memory when we were forced by space limitations to adjourn. We'll now pick up where we left off.

The process of loading the Disk Operating System into memory is often called booting the operating system, or booting DOS—or sometimes just booting the disk. Here's why.

The physical nature of the computer is such that when you turn the power on, the microprocessor takes its first orders from certain addresses in read-only memory. On the Personal Computer, those ROM-resident instructions cause an automatic hardware checkup to take place. Then, if you have a disk drive attached to your system, a small section of code stored in ROM causes the microprocessor to fetch whatever information happens to be located in a specific sector of the disk in drive A.

If that disk is a system disk (either DOS, a copy of DOS, or a disk formatted to be a system disk; see System Notebook, July 1982), then the information in that first-read disk sector causes the rest of the operating system, stored as a larger body of code elsewhere on the disk, to be loaded into memory.

This cascade sequence, in which a tiny bit of permanently resident code causes two successively larger chunks of instructions to become operative within the computer, has the net effect of bringing the system to its feet—of picking it up by its bootstraps. The ROM-resident code that initiates the process is called a bootstrap loader.

A couple of things are worth noting here. First, the computer keeps only the barest, most essential data in ROM. The bootstrap loader has to be in ROM because without it or something equivalent (like a switch on the outside of the machine), you'd have no way of communicating with the microprocessor. But the operating system, vital though it may be for most of your dealings with your computer, takes up quarters within the volatile, changeable portion of memory called RAM.

The fact that the operating system is housed in RAM means that you can extend it, alter it, or delete it—if you wish and you know how. Or you can substitute some entirely different body of software to perform equivalent functions.

There are in fact three major competing operating systems for the IBM Personal Computer- DOS (more formally called PC-DOS), CP/M-86, and the UCSD p-System. IBM markets them all, although they were written by other companies. (DOS is by Microsoft, CP/M-86 comes from Digital Research, and the p-System is licensed from SofTech Microsystems.) Other systems are available from other vendors. We've focused on DOS in these articles primarily for the sake of illustration.

Nifty though it may be to move data in and out of memory or to copy files from one disk to another, these are not the functions for which you bought your computer. Odds are you bought it for some more tangible purposes, like tracking your accounts receivable or your family budget. Or perhaps for word processing or games.

The kind of instructions that enable your computer to do these more concrete chores fall under the heading of application software, as distinguished from system software. The Samples programs on your DOS disk, which we ran in the first installment of this column, are relatively simple examples of application software.

You may remember, though, that in order to run the Samples programs you had to type the command *BasicA Samples*. In doing that, you loaded yet another large body of system software into memory.

The command *BasicA Samples* that you used in the first installment of this column is actually a shorthand way of issuing two commands at once. The two commands, if issued separately, would be *BasicA* and *Run "Samples*. We'll look at the first one, *BasicA*, first.

When you type *BasicA* and hit enter, you are said to be invoking the Basic interpreter. More exactly, what you're doing is issuing an instruction to DOS, asking the operating system to load into memory a file known formally as BasicA.com and informally as Advanced Basic.

Advanced Basic, like DOS, is a piece of system software. Specifically, it's a programming language, which means its function is to facilitate the creation of application software and provide an environment in which certain application software can be run. We'll unravel these words presently.

If you're "in DOS"—that is, if you're looking at the prompt A> —try typing the following:

```
Print 2+2
```

and hitting the enter key. The disk drive will come to life, and in a moment you'll receive a little message saying "Bad command or file name." DOS doesn't seem to get your drift.

DOS interprets your input as an instruction to load a file named Print 2+2.com or Print 2+2.exe or Print 2+2.bat. Such a file cannot exist (unless, perhaps, you're running some operating system other than DOS) because it violates DOS's file-naming rules. But DOS goes looking, just the same. And failing to find such a file, it sends an error message to the screen and returns the ball to you with another A>prompt.

Now, with the DOS disk in drive A, type *BasicA* and hit enter. This time DOS hunts for, finds, and loads into memory the file BasicA.com. As a consequence of its loading that file, you get some other messages on your screen. At the top of your screen you should see a title line, a version and copyright notice from IBM, and information about how much memory you have available. You'll also see some other information at the bottom of your screen.

Under the memory message you'll find the word Ok. That's a prompt, rather like the A>prompt you get from DOS. It's the computer's way of telling you that Advanced Basic has been loaded into memory and that it's your turn to talk to the computer. Now try typing the same line you offered to DOS, *Print 2+2.*

This time the response should be a little more gratifying. In fact, you've now brought your computer up to the functional level of a hand calculator.

If two plus two doesn't impress you, try typing *Print 5933/10971.* Again, you get a meaningful response. The computer knows what the numbers mean and recognizes the slash as an instruction to divide.

If typing the word *Print* is irksome to you, substitute a question mark. The effect will be the same as if you spelled out *Print.*

Now try typing *Write 2+2.* Surprise, it still works (the 4 will show up all the way on the left side of the screen instead of having a space on its left, but that's beside the point for now). If you're starting to feel intoxicated with power now, try *Compute the sum of 2 and 2 and display the results on screen.*

Boo, the computer is just as dumb as it was when you were trying to get DOS to calculate. The message you get this time, "Syntax Error," is roughly Basic's equivalent of "Bad command or file name."

It may be too obvious to mention, but it probably bears saying at least once in this column, that your computer is profoundly stupid. Even when it seems to be understanding you, when you and it have, for all appearances, established the most meaningful of relationships, there is nothing resembling understanding or cognition taking place within that system unit.

What your computer is, among other things, is an ingenious maze of electrical circuitry that occasionally, under precisely controlled circumstances, can take signals that mean something to you, transform them within itself, and send back other signals that also mean something to you. Basic is an intermediary that facilitates this process.

If you talk to Basic just right, it will translate your words into electrical impulses that will cause the microprocessor to do something more interesting than print "Syntax Error" on your screen. Basic's vocabulary is only a little over a hundred words, and it imposes some exasperating syntactical constraints upon your conversation; but despite those limitations it will allow you to do some fairly interesting things with your computer.

Earlier we said that Basic was a programming language. We need to clarify that statement, because so far nothing we've done could be construed as programming.

If you're still in Basic—that is, if you're looking at the Ok prompt—try typing *10 Print 2+2.* This time when you hit enter you should get neither a 4 nor an error message. All that appears to happen is that your cursor moves down to a new line.

Now type *run* and hit enter (or hit the F2 key; the effect is the same). When you do this you get your answer—4. This procedure—preceding the command with a number (we chose 10 arbitrarily; it could have been any positive whole number from 0 to 65529)—is still not programming, properly speaking, but it's different from what we were doing before.

What we just did was issue a command in the so-called deferred execution mode. Earlier, when we were giving orders that didn't start with numbers, we were operating in what's called the immediate execution mode.

Now type *20 Print 3+3*, hit enter, and then hit the F2 key again (or type *run* and hit enter). Now, at last, you've created a (rather primitive) program. The computer remembers the line you typed earlier, the one that started with 10, and when you type *run* it responds to both lines, starting with the one that begins with the lower number. If you had typed the line that started with 20 before typing the line that started with 10, it wouldn't have made any difference; the computer would sort them out internally and still respond first to the line that started with 10.

Now try typing *List* and hitting enter. There are your two lines again, just as you typed them originally (except that they now appear entirely in capital letters). Basic has stored the commands in memory and will dutifully screen them in response to the command *List.* Again, it wouldn't matter in what order they had originally been typed. *List* would present them in ascending numerical order.

One more example. If you've still got the Ok prompt, type *Load "Samples.* Hit enter. Now type *List* and hit enter and you'll see what a more typical Basic program looks like.

Basic is by no means the only way to create programs. But it's one of the more popular means of creating programs on personal computers. We'll have more to say about Basic and about programs in general when this saga continues in September. ▲

NF SYSTEMS

A new micro-software company with more than a decade of experience in large and intermediate systems.

— Catalog Available —

Presenting a series of programs designed to organize and control paperwork for the entire family.

MAILING LIST Print 1, 2, 3 up labels from a file sorted on Name or Zip Code **64.95**
If you already own Household Aids **49.95**

STOCK PORTFOLIO Keep all transactions by company name and date. Print entire portfolio or transactions by month or year **39.95**

CHECK REGISTER Up to 40 ledger/budget headings. Print Check Register or Totals by ledger/by month **39.95**

CROSS REFERENCE A must for basic programmers. List includes variables, constants, functions and sub-routines. **34.95**

SORT SUB-ROUTINE Call from any Basic program – sorts in ascending or descending order. **24.95**

DISKETTE CATALOG Create a master catalog of all your diskettes! If PC - catalog is created from disk. For other diskettes, enter from keyboard. May change and/or search for a particular file. **A Must!** **29.95**

HOUSEHOLD AIDS Includes Household Inventory, Personal Statement, VCR Index, Resume, Name/Address/Phone. **49.95**

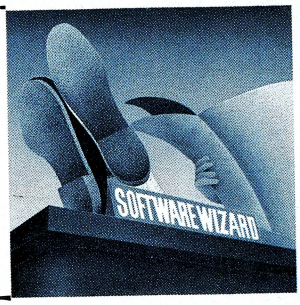
NF SYSTEMS, LTD.
P.O. Box 76363
Atlanta, GA 30358

VISA

IBM* Personal Computer

To order, call collect 9-5 404-252-3302

tradetalk



Δ **Eagle Software Publishing** (Wayne, PA) has sold 16,500 shares of common stock in a private offering through the underwriters Blank, Conger & Sena, of Cincinnati, Ohio. The money will be used to hire product developers and finance equipment. Eagle specializes in business software, notably the program *Money Decisions.*

Δ **The Source** (McLean, VA) has announced *Participate*, a computer conferencing service. Conferences can cover any subject, and they can be open to the general public or closed. You can also transcribe the conference to printer. *Participate* is available to all Source users at regular rates. Meanwhile at the Source, **Bettie Alexander Steiger** has been appointed vice president for corporate relations. One of the original founders of the Source, Steiger served previously as vice president of information resources. Her old position will be assumed by **Jane Brown**, formerly manager of database acquisition.

Δ **Leading Edge Products** (Canton, MA) has created a research and development division. The new division is headed by **Phil Florence**, executive vice president. Leading Edge has also announced an exclusive distribution agreement with **BMC, USA,** the American arm of Business Machines Corporation of Japan. Leading Edge is now the sole distributor of all BMC monitors in the United States and Canada. In addition, Leading Edge has named **John Fisher** as manager of technical services. Before coming to Leading Edge, Fisher was a field engineering manager at Honeywell.

Δ **Randy Knox** has been appointed vice president of manufacturing at **Davong Systems** (Mountain View, CA). Knox comes from Versatronex and earlier worked at Intel.

Δ **VisiCorp** (San Jose, CA) has entered an agreement with **Data Resources, Inc.** (Lexington, MA), to make DRI's extensive business and economic data bank available to Personal Computer users. The two companies have agreed to develop a program that will allow pc owners access to the DRI data in the form of preformatted *VisiCalc* models. The product is planned for release in the third quarter of 1982.

Δ **Thomas Kindel** has been appointed regional sales manager for the western states for **Amdek Corporation** (Arlington Heights, IL). Kindel was formerly southern California sales manager for Apple Computer, and earlier worked for Motorola and then Heath/Zenith. Amdek has a line of video monitors and peripheral equipment for the pc.

Δ **Digital Research** (Pacific Grove, CA) has opened a regional sales office for the eastern region—Connecticut, Massachusetts, Maine, New Hampshire, New Jersey, New York, Pennsylvania, Rhode Island, and Vermont—in Burlington, Massachusetts. The office will furnish education services and customer support, in addition to sales. **Bruce Cohen** is joining Digital as regional manager. Cohen was previously northeast regional director for Datapoint.

Bill Gates, of Microsoft, has moved from administrative duties back into research.

Δ **Bill Gates** has stepped aside as president of **Microsoft** (Bellevue, WA) in order to direct planning and technical research more closely. Replacing Gates as chief operating officer is **James Towne**, who comes to Microsoft from Tektronix, where he headed the instruments division. Microsoft spent six months searching for a new president before choosing Towne. ▲

The Best Things in Life

How many things in life can you get free?

Phone books, lousy television programming, advice, criticism, bad weather, and lunch if you're lucky; that's about it.

There's one other thing you can get free if you own an IBM Personal Computer—the magazine you're holding. For a full year we'll send you a magazine packed with useful information about the pc free of charge.

Beginners' Corner, The Profit Plot, System Notebook, Marketalk News, Marketalk Reviews, Newspeak, Best Sellers—each month Softalk for the IBM Personal Computer will help you learn about the pc with regular features and timely applications articles.

If you know someone who owns an IBM pc and who doesn't receive our magazine, send in that person's name, address, and serial number of their pc and we'll add an extra free month to your trial subscription. Such a deal!

softalk
for the IBM Personal Computer

Circulation
11021 Magnolia Boulevard
North Hollywood, CA 91601

Software Worthy of the IBM-PC

$129.95
for *IBM-PC version,
or Apple II or III.

Why did you buy an IBM-PC rather than some other brand of computer? We think we know the answer. You trust the name IBM and the quality it stands for.

Why should you select **WRITE-ON!** as your word processing program? Because it's totally worthy of your IBM-PC. BYTE Magazine, for example, said things like: "... human engineering is superb ... commands are easy to learn and use ... the most powerful features found in a microcomputer-based word processor ..." Beyond this, **WRITE-ON!** is fully supported by Datamost, Inc., a sound and solid company of professional specialists.

WRITE-ON! can handle all your word processing whether simple or complex. You can set tabs, margins, spacing, with a keystroke – you can move, copy, merge, delete or save blocks of text. And you can save time with its form letter or its complete mail-merge features. What about editing your letters or documents? We've made that especially easy and uncomplicated, by making sure you can use the regular IBM edit keys.

You'll also appreciate **WRITE-ON's** capability as a powerful program editor. Nothing near its price edits BASIC, PASCAL, FORTRAN, COBOL, and Assembly Language source code so well. You can even use it to edit VISICALC and accounting system files.

All in all **WRITE-ON!** is the word processor you can have confidence in ... the one that delivers the reliability and usefulness that make it truly worthy of your personal computer. And, with all this, it is exceptionally inexpensive. Get your copy now from your computer store, or from:

DATAMOST INC

9748 Cozycroft Ave., Chatsworth, CA 91311 (213) 709-1202

*IBM-PC a trademark of IBM. Apple II & III are trademarks of Apple Computer, Inc.

VISA/MASTERCHARGE accepted. Add 3% for shipping/handling. (California residents, add 6% sales tax also.)

softalk presents the bestsellers

The expected influx of new software into the IBM Personal Computer marketplace failed to materialize in June, leaving this Bestseller report a rather ho-hum affair. *VisiCalc* and *WordStar* are selling orders of magnitude more units than any other software, and the rest are falling in behind.

The dominance of the two programs has led to a minor change in the methodology of calculating the index that appears in the Top Thirty listing. Rather than setting the second-place program's index at approximately 100 and calculating every other program from that point, this month the third-place program was set to nearly 100. The index number still functions the same way, but the index column now better visually reflects the marketplace, with the top two programs standing out.

Only three new programs made their way onto the list in June. *VideoTrek 88*, a real-time Star Trek genre game, scored at twenty-first. *EasyWriter II*, the version being marketed directly by Information Unlimited Software, was twenty-fifth. *MBA* from Context Management Systems was twenty-sixth.

MBA was the only one of the new entries to be in its introductory month in the marketplace and is a case study of changing marketing trends.

For the most part, microcomputer programs have been designed to be "user friendly," a term that seems to cover a multitude of sins of omission and commission. One of the implications of this drive for turnkey software is simplicity of purpose. Even if memory limitations did not exist, most software publishers would find it counterproductive to attempt too much with one package.

Once a software package gets into loading separate modules and disk-swapping, past experience indicates, the end-user confusion factor increases by at least the third power of the number of such activities involved. End-user confusion translates directly into high cost of after-sale support on the part of both the publisher and the retailer. The path of least resistance has been to create several programs, each fairly trivial to learn and get running.

Now comes Context Management with a recognition of the difficulties of overloading the end user, yet with a determination to overcome those obstacles. *MBA* is an electronic speadsheet, word processor, and more—all in one package. You need 256K to run the program and you probably cannot hop down to your local retailer and buy it off the shelf.

The company is carefully hand-picking the dealers who will have the package available and giving each dealership a thorough orientation to the software before releasing it for sale. The limited number of dealerships already on the program makes it unlikely that it's available in your area yet.

But those dealers with the product in stock in June had a heyday. Literally every unit they received from Context Management went out to an end user. Regardless of whether Context Management's concept ultimately succeeds, the entire microcomputer industry should benefit from the experiment.

Software sales were fairly brisk again in June, although the lack of new product depressed the market some. The arrival of *VisiDex* and *VisiFile* from VisiCorp and sundry other packages should give the market a boost.

Even though the same programs made the Top Thirty, there was significant juggling in position.

Home Accountant Plus rose from fourth to third, *Deadline* moved from seventeenth to fourth, and Peachtree's *General Ledger* accounting package was fifth. *SuperCalc* rose from ninth to sixth, *MailMerge* jumped from a tie for seventeenth to seventh, *Asyn-*

We think our new mailing list program is the best in the world. And we're not alone.

1st CLASS MAIL. By Bob Schoenburg and Steve Pollack. Here's a pair of authors who may be software's answer to Irving Wallace. Wallace is the bestselling author who hears the murmuring of general populace and caters to their desires. Bob Schoenburg and Steve Pollack seem to have the same trait.

Consider.

The pair brought out *Home Money Minder*, a perfectly respectable home finance package. Then they listened to the user feedback. The result was *Home Accountant*, one of the phenomenal success stories of the first half of 1982.

Around the same time *Home Money Minder* hit, the team also tested the market with *The Mailroom*. Mailroom never was the success of *HMM*, but the authors used the same technique – listen to the users and incorporate all the good ideas. The result is **1st CLASS MAIL** – a program that, incredibly enough, manages to live up to its double-entendre name.

All of the above is not to accuse Schoenburg and Pollack of putting out the programming equivalent of stalking-horses to do their market research for them. Their original efforts do stand on their own merit. They just pale next to the sequels.

1st CLASS MAIL is so well thought out and so easy to use that other publishers who call their programs "user friendly" should bow their heads in shame.

SOFTALK got an early release of the program sans documentation. Yet a rank computer illiterate was able to apply the program to two separate uses with relatively little trouble. This is high praise indeed: that a novice operator could use a powerful program with no more than the screen menus.

The program allows for twelve fields, clearly more than the traditional name and address of a mailing label. The implication is that the software can be put to other innovative uses as well. The built-in ability to sort and filter on any field or combination thereof enhances the chances that users will find multiple applications for the program.

Continental Software will actually be publishing four versions of the program.* The one already in release is for the Apple II using floppy disks. A hard-disk version will follow. Both versions are pending for Apple III as well, awaiting the development of a rapid binary sort subroutine.

1st CLASS MAIL is a first-rate program for specialized data base applications.

Reviewed by Al Tommervik, Publisher, Softalk.

***Available for: Apple II,™ Apple III™/III with Profile,™ IBM-PC™/IBM-PC with Tecmar™ hard disk/IBM-PC with Davong™ hard disk.**

Continental Software

Continental Software, 11223 South Hindry Avenue Los Angeles, CA 90045 Telephone (213) 417-8031

Copyright 1982 by Softalk Publishing, Inc. Reprinted with permission.

Apple II, Apple III/III with Profile are registered trademarks of Apple Computer, Inc. Tecmar is a registered trademark of Tecmar, Inc. Davong is a registered trademark of Davong Systems, Inc. IBM-PC is a registered trademark of IBM Corp. 1st Class Mail, Home Money Minder and Home Accountant are registered trademarks of Continental Software.

chronous Communications Support from IBM and *Zork I* tied for eighth, and *Total Information Management* rounded out the top ten.

BPI's *General Accounting* package leapfrogged from twenty-fifth to eleventh, the largest jump of the month. *Microsoft Adventure* dropped from third to twelfth, tied with *IBM Pascal* and the *Fortran Compiler*. *Temple of Apshai* maintained sixteenth, *Zork II* was seventeenth, and *Space Guardian* was eighteenth. *EasyWriter* in its IBM form, both 1.0 and 1.1, was nineteenth, and *Typing Tutor* was twentieth.

The appetite for spreadsheet programs is phenomenal, with *VisiCalc*, *SuperCalc*, and *MBA* all on the list, although neither of the Johnny-come-latelies seem to impact on *VisiCalc*'s sales.

The word processing application is showing some interesting divergence. Among software buyers who buy outside of IBM authorized dealers, *Volkswriter* is the clearly preferred word processor. But few IBM dealers carry the product. Another facet of that same application is the proofreading program, where *SpellGuard*, *SpellStar*, and *EasySpell* seem to be dividing the market.

Some dealers are still reporting shortages of *Macro Assemblers* and *Fortran Compilers*. It's a commentary on the difficulty of serving the marketplace that product produced by one of the largest companies in the industry (Microsoft) for the largest company in the industry (IBM) can't seem to find its way easily through the distribution channels.

If the creators of the marketplace can't make it function, it's no wonder that such folks as Context Management are developing their own means of distribution. Even if their program were trivial, it's apparent that the distribution networks serving other microcomputer markets are not functioning in the IBM market, making it necessary for the independents to find other ways. ▲

IBM Personal Computer sales outlets representing approximately 10.6 percent of all sales of pc and pc-related products volunteered to participate in the poll.

Respondents were contacted early in July to ascertain their sales for the month of June.

The only criterion for inclusion on the list was the number of units sold—such other criteria as quality of product, profitability to the computer store, and personal preference of the individual respondents were not considered.

Respondents in July represented every geographical area of the continental United States.

Results of the responses were tabulated using a formula that resulted in the index number to the left of the program name in the Top Thirty listing. The index number is an arbitrary measure of relative strength of the programs listed. Index numbers are correlative only for the month in which they are printed; readers cannot assume that an index rating of 50 in one month represents equivalent sales to an index number of 50 in another month.

Probability of statistical error is plus or minus 2.7 percent, which translates roughly into the theoretical possibility of a change of 2.31 points, plus or minus, in any index number.

PC CONNECTION

YOUR LINK TO THE BEST SELLING SOFTWARE FOR THE IBM*PC

Data Base Manager	Alpha Software
Mailing List	
Question	
Type Faces	
The Apple—IBM Connection	
Easy Writer II	Information Unlimited Software
Easy Speller	
Easy Filer	
Home Accountant Plus	Continental Software
Las Vegas Blackjack	Quala
Space Guardian	Omric
SuperCalc	Sorcim
Temple of Apshai	Automated Simulations
Far Reaches of Apshai	
Curse of Ra	
VisiCalc	Visi Corp
VisiDex	
Desk Top/PLAN	
Volkswriter	Lifetree
Wordstar	Micro Pro
Mailmerge	
Zork I	Infocom
Zork II	
Deadline	

For more top thirty and other software for the IBM*PC
Call or write

PC CONNECTION

PC Connection ■ 6 Mill Street ■ Marlow, NH 03456 ■ (603) 446-3383
HOURS: 9:00 to 5:30
Visa and Mastercard Accepted
COD arranged
We will protect you against product defect for 30 days
*REGISTERED TRADEMARK OF INTERNATIONAL BUSINESS MACHINES

the top thirty

This Month	Last Month	Index	
1.	1.	510.38	**VisiCalc,** Software Arts/Dan Bricklin and Robert Frankston; VisiCorp, IBM
2.	2.	397.27	**WordStar;** MicroPro
3.	4.	99.31	**Home Accountant Plus,** Mike Farmer, Bob Schoenburg, Larry Grodin, and Steve Pollack; Continental Software
4.	17.	97.94	**Deadline,** David Lebling and Marc Blank; Infocom
5.	8.	93.80	**General Ledger;** IBM (Peachtree Software)
6.	9.	89.66	**SuperCalc;** Sorcim
7.	17.	86.90	**MailMerge;** MicroPro
8.	5.	77.25	**Asynchronous Communications Support;** IBM
	14.	77.25	**Zork I,** David Lebling and Marc Blank; Infocom
10.	13.	71.73	**Total Information Management;** Innovative Software
11.	25.	64.83	**General Accounting,** John Moss and Ken Debower; BPI
12.	3.	63.45	**Microsoft Adventure,** Gordon Letwin (Will Crowther and Don Woods); IBM (Microsoft)
	12.	63.45	**IBM Pascal,** Microsoft; IBM
	19.	63.45	**Fortran Compiler,** Microsoft; IBM
15.	7.	59.31	**Macro Assembler,** Microsoft; IBM
16.	16.	56.56	**Temple of Apshai,** Jeff Johnson, John Freeman, and J.W. Connelly; Automated Simulations
17.	20.	55.18	**Zork II,** David Lebling and Marc Blank; Infocom
18.	10.	53.80	**Space Guardian;** Omric
19.	6.	49.66	**EasyWriter 1.1,** John Draper; IBM (Information Unlimited Software)
20.	15.	41.38	**Typing Tutor,** Michael Sierchio (Dick Ainsworth and Al Baker); IBM (Microsoft)
21.	—	38.62	**VideoTrek 88;** Windmill Software
	21.	38.62	**UCSD Pascal,** Kenneth Bowles; IBM (SofTech Microsystems)
23.	23.	37.24	**VideoGraph 88;** Windmill Software
24.	27.	33.11	**Accounts Receivable;** IBM (Peachtree Software)
25.	—	31.73	**EasyWriter II,** John Draper; Information Unlimited Software
26.	—	28.97	**MBA;** Context Management Systems
27.	22.	26.21	**Time Manager,** Dick Ainsworth, Al Baker, and Jeffrey P. Garbers; IBM (Microsoft)
	11.	26.21	**Las Vegas Blackjack,** Charles Montante; Quala
29.	28.	24.83	**Inventory;** IBM (Peachtree Software)
30.	29.	23.45	**Accounts Payable;** IBM (Peachtree Software)

As IBM would design it

THE PERSONAL HARD DISK™

Only for the IBM Personal Computer

from Corona™

Corona experience makes the difference:

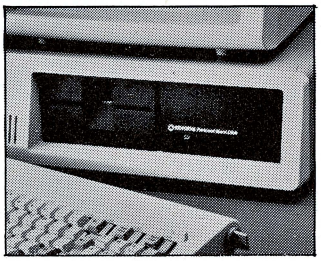
It fits right in—right inside the IBM Personal Computer, with no external power supply.

And it makes friends fast—with self-documenting, menu-driven, very "user-friendly" software.

It responds to your needs—for multiple operating systems on the same disk at the same time, in flexible partitions that grow as needed.

It backs you up—on floppies, using "selective" backup (by date, by volume, by category), so an additional backup device is not needed.

And it has reliability you can count on—combining exclusive use of Seagate drives with Corona's DataGuard™ and FailSafe.™



And the price?

$1995 for 5 MB,

$2495 for 10 MB

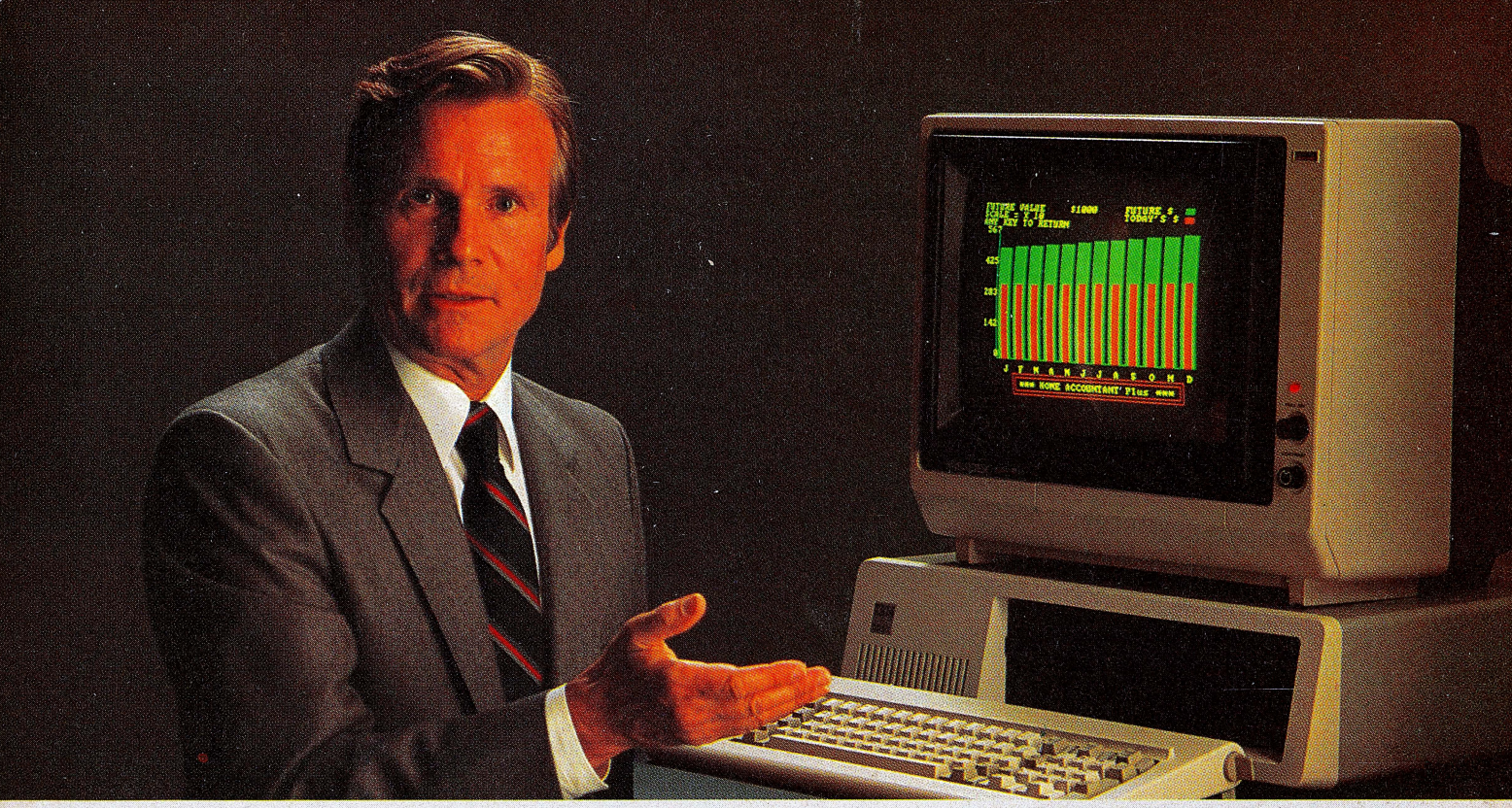
Experience the difference experience makes.

Ask your dealer to show you the Corona Personal Hard Disk (or call us for dealership and quantity purchase information).

corona

The Third Generation Microcomputer Company

Corona Data Systems 31324 Via Colinas, Section 110, Westlake Village, CA 213-706-1505



IBM OWNERS ARE DISCOVERING WHAT OVER 10,000 APPLE OWNERS ALREADY KNOW.

Chances are you don't truly realize how helpful your computer can be in managing your family finances.

The key is selecting the right software. That's why you should know about a program that's become the runaway best seller among Apple owners* — the Home Accountant™ from Continental Software.

ADDED POWER FOR IBM USERS. Home Accountant Plus for the IBM Personal Computer has all the power and ease of operation that made the Apple version such a success. But it includes a brand new Forecasting Module that gives you a peek at the future. You can evaluate various investment strategies — or let the program show the way to your financial goals.

Of course, you get all the great features of Home Accountant. Like room for up to 100 budget categories, 5 different checking accounts, and all the credit cards you can carry. Automatic printing of your net worth and standard financial statements — plus checks and any custom reports that you specify. Also, full-color graphs† of actual vs. budgeted expenditures, trend line analysis, and more.

READY TO GO TO WORK—TODAY. Sooner or later, you're going to want your computer's help with your personal finances.

Over 10,000 users have already found the ideal answer. Now it's your turn.

Home Accountant Plus. About $150 at computer stores everywhere.

Continental Software

Call or write us for your nearest dealer: Continental Software, 11223 Hindry, Los Angeles, CA 90045 (213) 417-8031. *Rated No. 1 in home software packages, Softalk magazine (March-June 1982). †Requires 64k, 1 disk drive and mono or color display. Printer optional. Apple is a registered trademark of Apple Computer Inc. IBM is a registered trademark of IBM.